AMSAA

TECHNICAL REPORT NO. 368

INTERACTIVE GRAPHICS PLOTTING SYSTEM (IGPS)

BRADLEY M. LUFKIN

SEPTEMBER 1982

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

U. S. ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
ABERDEEN PROVING GROUND, MARYLAND

DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the DARCOM position unless so specified.

TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
Technical Report No. 368

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*
INTERACTIVE GRAPHICS PLOTTING SYSTEM (IGPS)

5. TYPE OF REPORT & PERIOD COVERED
Technical

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR*(e)*
Bradley M. Lufkin

8. CONTRACT OR GRANT NUMBER*(s)*

9. PERFORMING ORGANIZATION NAME AND AODRESS
US Army Materiel Systems Analysis Activity
Aberdeen Proving Ground, MD 21005

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Commander
US Army Materiel Development and Readiness Command
5001 Eisenhower Avenue
Alexandria, VA 22333

12. REPORT DATE
September 1982

13. NUMBER OF PAGES
79

14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*

15. SECURITY CLASS. *(of thie report)*
UNCLASSIFIED

15a. OECLASSIFICATION/DOWNGRADING SCHEOULE

16. DISTRIBUTION STATEMENT *(of thie Report)*

Approved for public release; distribution unlimited.

17. OISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

The term "he" (and its derivatives) when used in this publication represent both the masculine and feminine genders; exceptions will be noted.

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Two-Dimensional Graphics
Graph Paper
Probability Paper
Weibull Paper
User-Definable Paper
Tektronix 4051

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

IGPS is a computer-graphics program which produces high-quality plots of two-dimensional data; it is written in BASIC for the Tektronix 4051 desk-top computer. IGPS features: a large variety of axis systems, including probability paper for several distributions, logarithmic paper, and user-definable paper; neat tick mark and axis labeling with automatic or user-controlled decimal-point selection; several curve marker and dashed line types; legends; curve fitting; plotting of user-defined functions; data storage and retrieval on all Tektronix 4051 peripherals; and storage and retrieval of entire plots.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

CONTENTS

Next page is blank

INTERACTIVE GRAPHICS PLOTTING SYSTEM (IGPS)

1. INTRODUCTION

IGPS (Interactive Graphics Plotting System) is a computer-graphics program, written in BASIC for the Tektronix 4051 desktop computer, which generates high-quality plots of two-dimensional data; the plots can be used for slide presentations or can be inserted into reports.

The program is interactive and prompts the user for all the information required to construct a plot; furthermore, the program is structured to permit rapid changes to an existing plot. Therefore, the user can keep modifying the plot he is drawing and can preview it on the screen of the Tektronix 4051 until he is satisfied with its appearance. Then, simply by depressing one key on the keyboard, the user can exactly reproduce his plot on a Tektronix 4662 x-y plotter.

Figure 1.1 is a sample plot produced in this fashion by IGPS on a 4662. The plot was drawn directly onto a piece of report paper 8-1/2 inches high by 11 inches long placed in the lower left-hand corner of the 4662. The figure number, title, and page number were then added with a conventional typewriter.

The sample, which could have been used directly to make a slide, displays some of the features of IGPS: an unusual axis system (in fact, Rayleigh Paper), neat tick mark and axis labeling, multiple curve marker types, legends, and curve fitting. Table 1-1 is a summary of the features currently available with IGPS.

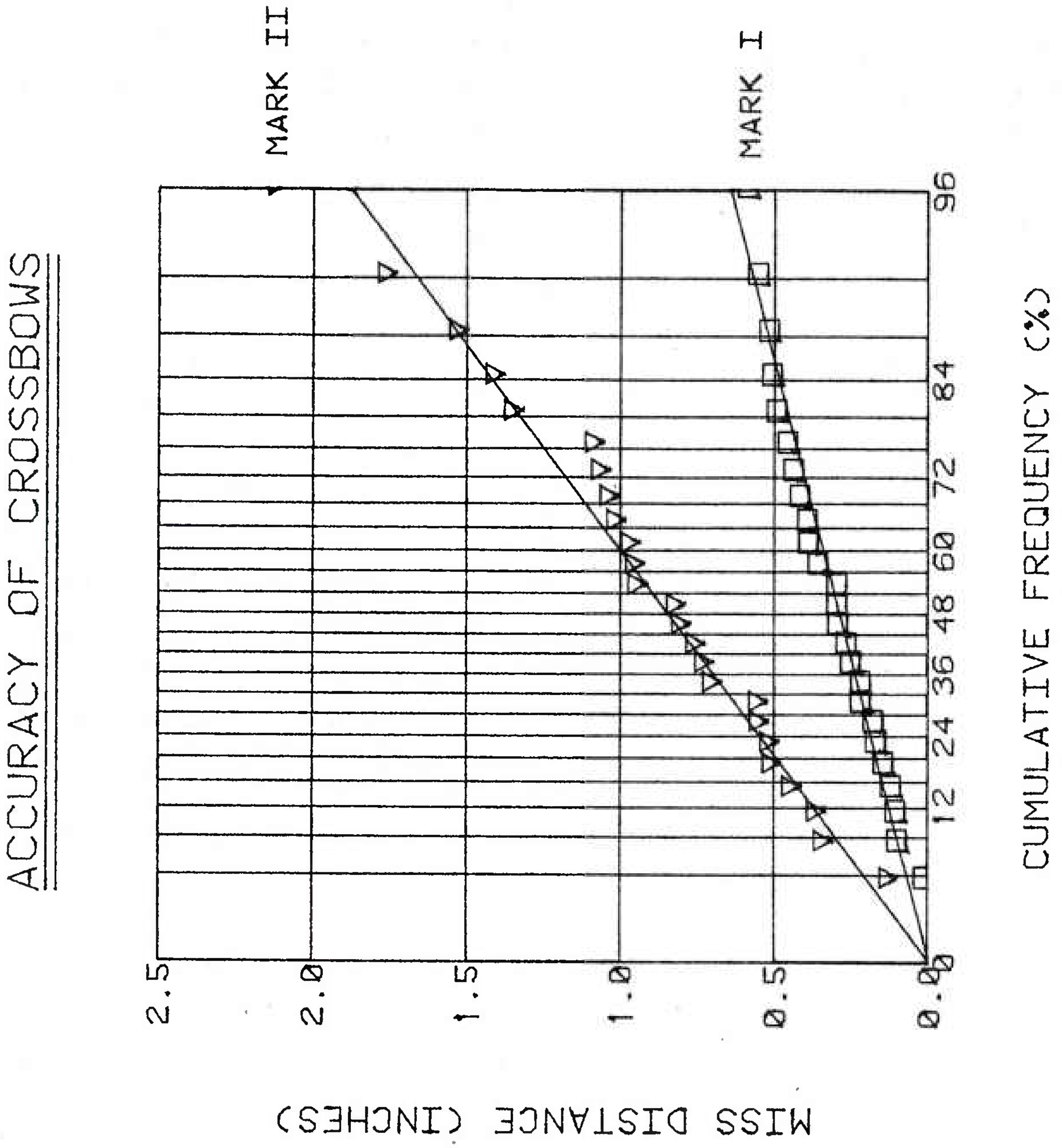

Figure 1.1 Sample Plot Produced by IGPS

TABLE 1-1 SUMMARY OF IGPS

Tick Mark Spacing:

- uniform
- user-controlled

Tick Mark Labeling:

- uniform with automatic decimal-point selection
- user-controlled

Frame Types:

- axis
- grid
- border
- suppressed

Standard Paper Types:

- linear in x and y
- linear in x, logarithmic in y
- logarithmic in x, linear in y
- logarithmic in x, logarithmic in y
- user-definable

Probability Paper Types:

- uniform distribution
- normal distribution
- lognormal distribution
- Rayleigh distribution
- Weibull distribution
- user-defined distribution

Input/Output Devices:

- Tektronix 4051 Internal Tape Drive
- Tektronix 4924 Auxiliary Tape Drive
- Tektronix 4907 File Manager (Disc)

Line Types:

- solid
- dashed
- dashed-dotted
- dotted

TABLE 1-1 SUMMARY OF IGPS (continued)

Curve Markers:

- crosses
- x-shaped
- squares
- triangles
- upside-down triangles

Curve Fits (Linear, Least-Squares):

- estimate slope and intercept
- estimate slope, intercept fixed
- estimate intercept, slope fixed

Plotting Devices:

- Tektronix 4051 Screen
- Tektronix 4662
- Tektronix 4662, Option 31

2. USING IGPS

Once IGPS has been loaded into the memory of the Tektronix 4051, it needs to be initialized. This can be done by typing in RUN followed by a carrige return. IGPS will initialize itself and will display the following menu on the screen of the 4051.

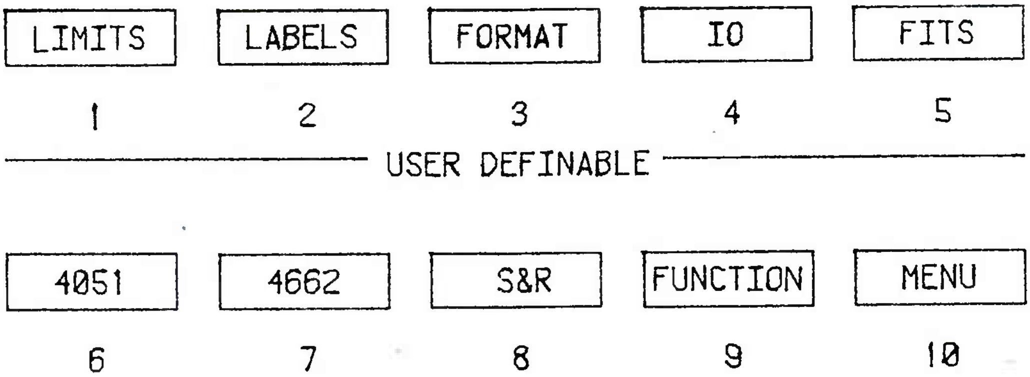

Figure 2.1 Menu for IGPS

The 10 rectangles in the menu correspond to the 10 user-definable keys located in the upper left-hand corner of the keyboard of the 4051. These 10 keys are used to control IGPS.

If, at any time during a plotting session with IGPS, the user wishes to examine the menu, he can either type in RUN followed by a carriage return or he can depress user key number 10.

Table 2-1 summarizes the action of the 10 user-definable keys.

TABLE 2-1 SUMMARY OF USER KEYS

User Key Number	Function
1	Establish limits over which plotting is done.
2	Inputs axis labels and title.
3	Defines axis system.
4	Inputs data curves from peripheral or keyboard; saves data curves on peripheral.
5	Does least-squares, linear fits on data curves.
6	Plots graph on screen of 4051.
7	Plots graph on 4662 x-y plotter.
8	Stores graph on periperal; retrieves previously saved graph from peripheral.
9	Generates x-y data from user-defined parametric functions.
10	Displays menu on the screen.

2.1 User Key Number 1: Limits.

This key is used to establish the limits between which data plotting is to take place (in other words, data outside the limits will not be plotted); the key is also used to define the distance between tick marks and between tick mark labels.

After the user depresses key number 1, IGPS will prompt him with

```
UNIFORM TICKS ON THE X AXIS (1=YES, 2=NO):
```

If the user responds with a 1, meaning that he wants uniformly spaced tick marks and labels, IGPS will prompt with

```
XMIN:
XMAX:
MAJOR TICKS:
MINOR TICKS:
```

The user should respond to each prompt with a suitable number. The response to XMIN establishes the smallest value on the x axis and that to XMAX the largest. The response to MAJOR TICKS establishes the distance between adjacent tick mark labels and the response to MINOR TICKS the distance between adjacent tick marks. The sequence to generate the x axis of Figure 1.1, for instance, is

```
XMIN: 0
XMAX: 96
MAJOR TICKS: 12
MINOR TICKS: 4
```

In no case should XMIN ever be the same as XMAX and, for best results, MAJOR TICKS and MINOR TICKS should be multiples of the difference between XMAX and XMIN; furthermore, MAJOR TICKS should be a multiple of MINOR TICKS. Results are unpredictable if these conditions are not met. In fact, IGPS will create n tick marks, where

$$n = 1 + [(\mathrm{XMAX} - \mathrm{XMIN}) / \mathrm{MINOR\ TICKS}] \quad ,$$

and the quantity in brackets denotes "the largest integer not to exceed," and the tick marks will appear at points x_i such that

$$x_i = \mathrm{XMIN} + \mathrm{MINOR\ TICKS} \cdot (i-1),\ i=1,\ldots,n.$$

Similarly, IGPS will create m tick mark labels, where

$$m = 1 + [(\mathrm{XMAX} - \mathrm{XMIN}) / \mathrm{MAJOR\ TICKS}] \ ,$$

and the labels will appear at points l_i such that

$$l_i = \mathrm{XMIN} + \mathrm{MAJOR\ TICKS} \cdot (i-1),\ i=1,\ldots,m.$$

Clearly, if MAJOR TICKS is the same as MINOR TICKS, then all tick marks will be labeled.

Consider the sequence

```
XMIN: 0
XMAX: 1
MAJOR TICKS: 0.25
MINOR TICKS: 0.25
```

This produces the following x axis

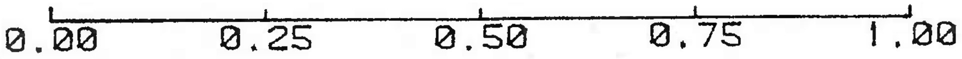

Figure 2.2 Sample Axis with Uniform Tick Marks and Labels

Note that the labels are drawn centered under the appropriate tick mark, and that all labels have the same number of digits (two) to the right of the decimal point. Note also that two digits beyond the decimal point is the ideal number in this situation, in the sense that using any fewer would have caused a loss of accuracy while using any more would have caused unnecessary trailing zeroes. IGPS calculates and uses this ideal number automatically.

If the user does not like uniformly spaced tick marks and tick mark labels with automatic decimal-point selection on the x axis, then he should respond with a 2 to the prompt

```
UNIFORM TICKS ON THE X AXIS (1=YES, 2=NO):
```

By doing so, he assumes total control over the placement and appearance of tick marks and labels; after such a response, IGPS prompts with

```
NUMBER OF TICK MARKS (AT LEAST 2):
LOCATION OF TICK MARKS:
NUMBER OF TICK MARK LABELS:
LOCATION OF TICK MARK LABELS:
```

The meaning of these prompts is obvious.

At least two tick marks are required since the first one entered is assumed to be the minimum value along the x axis while the last one is assumed to be the maximum value. The number of tick mark labels, however, is unrestricted and may even be zero; if it is zero, the fourth prompt is not displayed since it is not needed.

The locations of tick mark labels must all be entered on one line and must be separated by a single blank or by a comma. The labels will be plotted by IGPS exactly as they were entered (in other words, IGPS will not calculate the number of places beyond the decimal point for the labels but will use as many as the user himself typed in).

For example, the sequence

```
NUMBER OF TICK MARKS (AT LEAST 2):  19
LOCATION OF TICK MARKS:  .1  .2  .3  .4  .5  .6  .7  .8  .9
                          1   2   3   4   5   6   7   8   9
                          10
NUMBER OF TICK MARK LABELS:  3
LOCATION OF TICK MARK LABELS:  .1  1  10
```

if used in conjunction with a logarithmic x axis, will produce

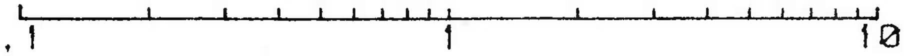

Figure 2.3 Sample Axis with User-Controlled Tick Marks and Labels

Once the user has supplied the specifications for the x axis, IGPS will prompt for the y axis in exactly the same way as it did for the x axis.

2.2 User Key Number 2: Labels.

This key is used to establish the axis labels and the title of the plot. If the user does not want these items, he should respond with a carriage return when prompted for them.

The x-axis label is plotted below the plotting area and is centered between the x-axis limits. The title of the plot is similarly centered but is plotted above the plotting area and is underlined twice (if the plot has no title, no underlining takes place). The y-axis label is plotted to the left of the plotting area and is centered between the y-axis limits. When plotted on the 4662 x-y plotter, the y-axis label will be written along a baseline rotated 90 degrees from the horizontal (see Figure 1.1); on the screen of the 4051, however, the individual characters are plotted on the horizontal, with the first character above the second, the second above the third, and so on (as in Chinese or Japanese)--this is done because the character-generator of the 4051 cannot produce rotated characters. Close examination of Figure 1.1 will reveal that different character sizes are used in various parts of the plot; this feature is also only available on the 4662 because the character-generator of the 4662 can produce characters of arbitrary size, unlike the generator of the 4051.

2.3 User Key Number 3: Format.

This key is used to control the basic appearance of the graph: it determines the type of frame which will be used, the speed at which the graph will be drawn, the color of the frame and labels and, more importantly, the type of paper which will be used.

After the user depresses user key number 3, IGPS will ask for the type of frame the user wants. There are four possible answers: zero, meaning no frame or labels are to be drawn; one, meaning draw an axis system; two, meaning draw a grid; and three, meaning draw borders.

The first option is useful when a set of curves is to be added to a graph because it avoids re-plotting most of the graph.

The second option produces axes running along the left-hand side and the bottom of the plotting area.

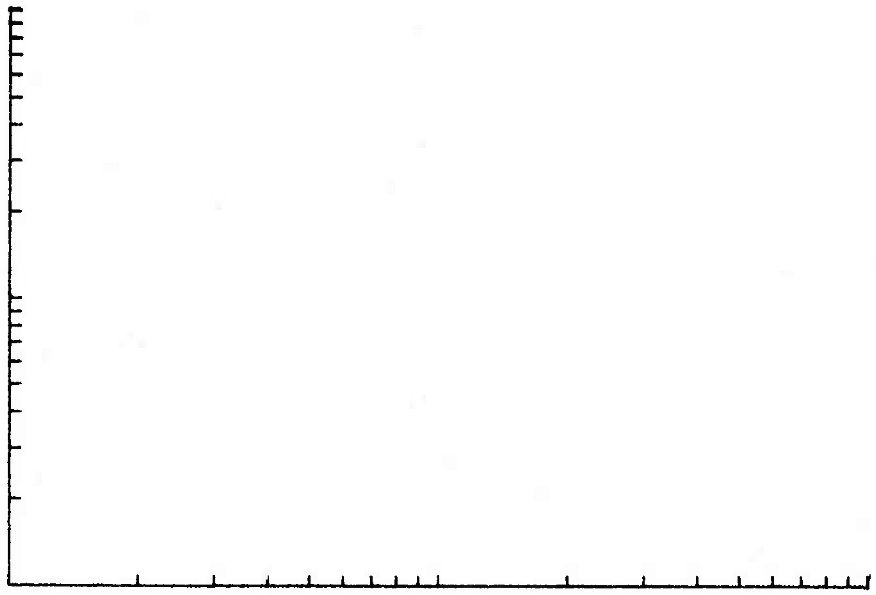

Figure 2.4 Sample Axis System

The third option produces a grid over the plotting area; each tick mark is indicated with a solid line.

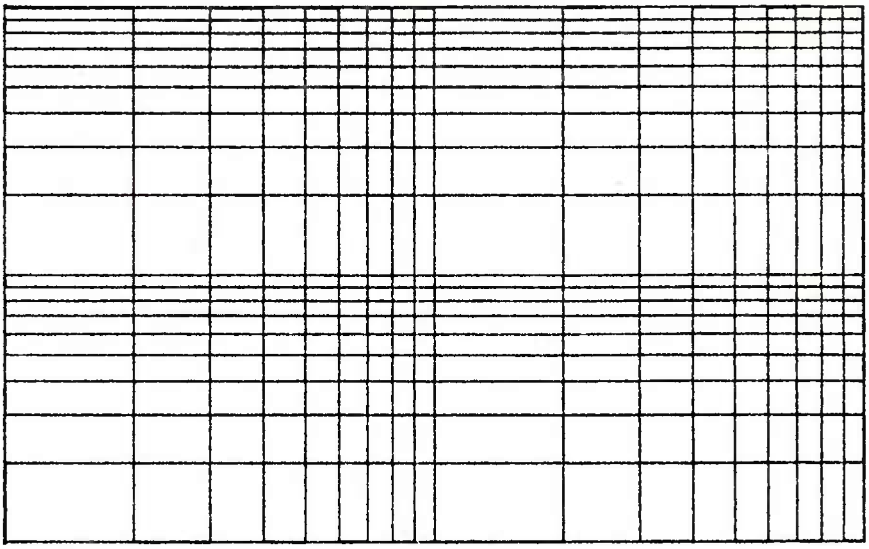

Figure 2.5 Sample Grid

The fouth option produces axes and tick marks along the edges of the plotting area.

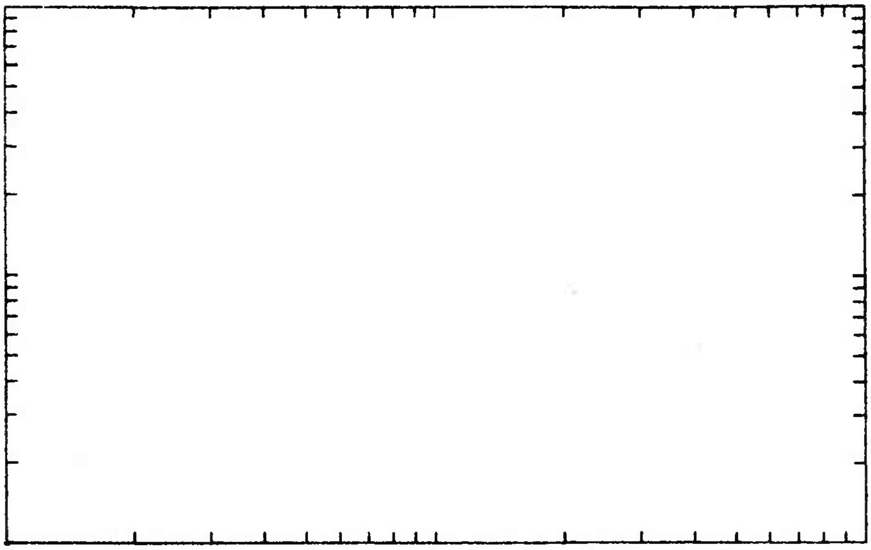

Figure 2.5 Sample Borders

After he selects a frame type, the user will be prompted for the color of the frame and labels. He must reply with an integer between one and eight, corresponding to the eight pens in the turret of the 4662 x-y plotter equipped with Option 31. The choice of color is ignored if plotting is either on the screen of the 4051 or on a standard 4662.

The user is then asked for a pen speed, to which he must respond with an integer between 10 and 570. The answer is interpreted in millimeters per second and is the maximum speed at which the 4662, Option 31, plotter will draw (the response is ignored on the 4051 or the standard 4662). For previewing, the user should select 570 mm/s. However, for best results, he should select 10 mm/s since the lower speed leads to much sharper definition of vectors.

The final prompt associated with user key number 3 is for the type of paper the user wants. IGPS will display

```
 PAPER TYPES:                        X-AXIS           Y-AXIS

 1.       LINEAR:                         X                Y
 2.         XLOG:                    LOG(X)                Y
 3.         YLOG:                         X           LOG(Y)
 4.       LOGLOG:                    LOG(X)           LOG(Y)
 5.         USER:                         ?                ?
 6.      UNIFORM:                     X/100                Y
 7.       NORMAL:                  N(X/100)                Y
 8.    LOGNORMAL:                  N(X/100)           LOG(Y)
 9.     RAYLEIGH:      SQR(-2*LOG(1-X/100))                Y
10.      WEIBULL:        LOG(-LOG(1-X/100))           LOG(Y)

 PAPER TYPE:
```

to which the user should respond with an integer between one and 10, corresponding to his choice of paper. As indicated by the prompt, the user's choice is in fact a selection of two transformations, or functions, one function for the x axis and one for the y axis. All data, including tick mark locations (but not tick mark labels) related to the x coordinate are transformed via the x-axis function before being plotted, as are all data related to the y coordinate via the y-axis function. Since the reader may not be fimiliar with all available paper types, we will describe each one, with its associated functions, separately.

2.3a Linear Paper. This is the most commonly used type of paper and is in fact plain graph paper. The two functions associated with this paper are the identity functions and therefore have no effect.

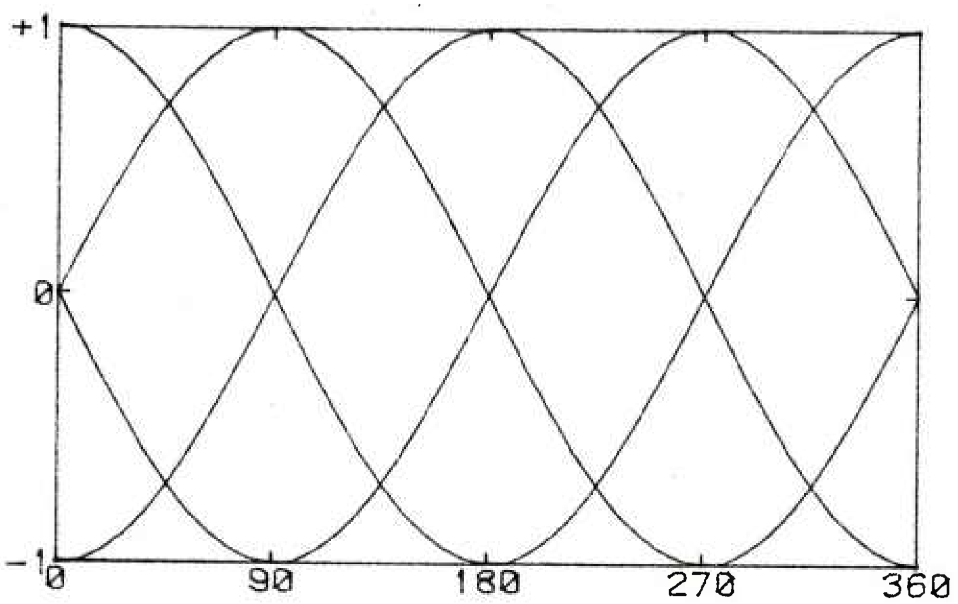

Figure 2.6 Example of the use of Linear Paper.

2.3b Xlog Paper. This paper has a logarithmic x axis and a linear y axis; therefore, all x-axis data are transformed via the function

$$F_a(x) = \ln x$$

before being plotted. Because of this transformation, there are implications for the meaning of the slope and intercept of any straight line which the user might choose to fit to his data while using xlog paper (Section 2.5, User Key Number 5). In particular, if we let a be the slope of such a line and if we let b be its intercept, and if the line appears to fit the user's data plotted on xlog paper, then an appropriate model of the user's data is:

$$y = a \ln x + b \ .$$

Though this may be perfectly obvious in this case, we have chosen to emphasize the point here because the situation becomes much more complex with other types of paper (transformations). In general, if we denote the two axis transformations by $F_a(x)$ and $F_b(y)$, then any straight line used with those transformations is in fact a model of the form:

$$F_b(y) = a \ F_a(x) + b \ .$$

If the user wants the model in simpler form, with only the y term appearing to the left of the equal sign, he will have to apply the inverse of F_b to both sides of the equation; this point is illustrated explicitly in the next section.

2.3c Ylog Paper. This paper has a linear x axis and a logarithmic y axis. To illustrate the use of this paper, consider the data in Table 2-2.

TABLE 2-2 VALUES OF AIR DENSITY AS A FUNCTION OF ALTITUDE

Altitude (km)	0	1	2	3	4	5	6
Air Density (kg/m^3)	1.22	1.11	1.01	0.91	0.82	0.74	0.66
Altitude (km)	7	8	9	10	11	12	13
Air Density (kg/m^3)	0.59	0.53	0.47	0.41	0.36	0.31	0.27
Altitude (km)	14	15	16	17	18	19	20
Air Density (kg/m^3)	0.23	0.19	0.17	0.14	0.12	0.10	0.09

From physical reasoning, or for some other reason, we decide that air density is an exponential function of altitude (in other words, that the logarithm of air density is a linear function of altitude). To test this hypothesis, we plot the data on ylog paper and fit a straight line with user key number 5.

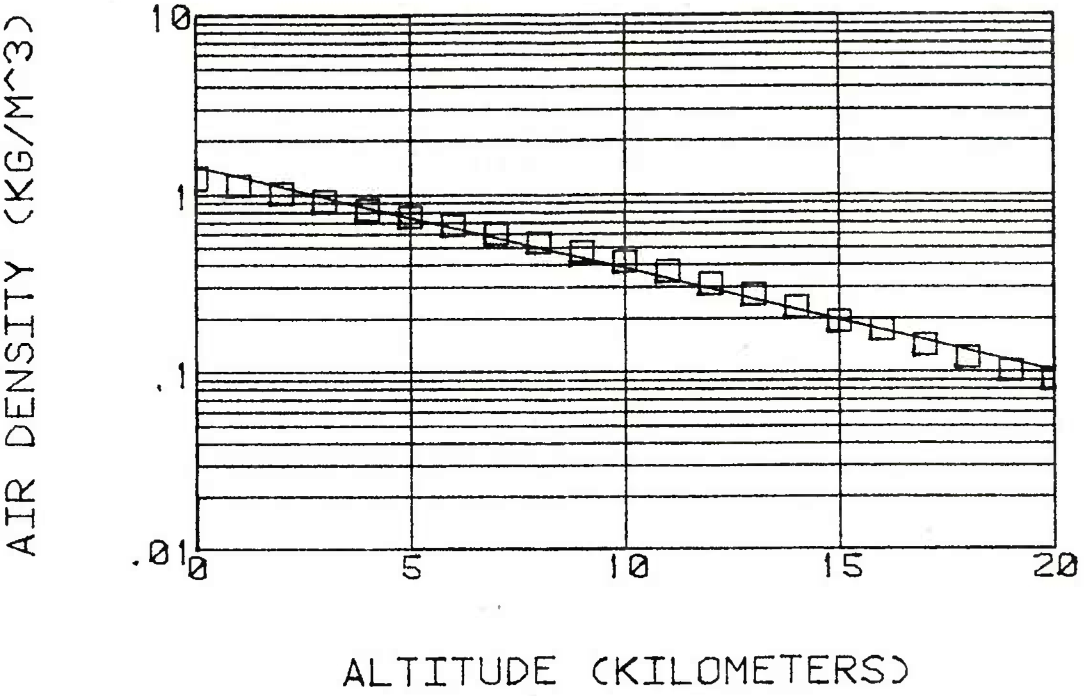

Figure 2.7 Example of the Use of Ylog Paper

The plot confirms our hypothesis. The slope and intercept of the line calculated by IGPS are:

$$a = -0.13,$$

and

$$b = 0.34 .$$

Therefore, our model of air density (p) versus altitude (h) is:

$$\ln p = -0.13\ h + 0.34 ,$$

or

$$p = 1.40\ e^{-0.13h} .$$

Note that this model does not go through the data point at zero altitude. Suppose we should want it to, in other words, suppose we should like a model of the form:

$$p = 1.22\ e^{-kh} .$$

what is the best value of k (in the least-squares sense)? Taking logarithms produces:

$$\ln p = -kh + 0.20 .$$

To determine k, we depress user key number 5 and request a straight-line fit with a forced intercept of

$$b = 0.20 \ .$$

IGPS will calculate the slope of that line as:

$$a = -0.12$$

It is clear, therefore, that the value of k we are looking for is:

$$k = 0.12 \ ,$$

and that the best model which goes through the value of air density at zero altitude is:

$$p = 1.22 \ e^{-0.12h} \ .$$

2.3d Loglog Paper. This paper has a logarithmic x axis and a logarithmic y axis. Therefore, straight-line fits to data plotted on this paper are models of the form:

$$\ln y = a \ln x + b \ .$$

The user is cautioned that whenever he is using a logarithmic axis, be it horizontal or vertical, he must avoid zero or negative values along that axis, both in the data being plotted and in the choice of tick marks. This is because the logarithm of a non-positive number is undefined.

2.3e User-Definable Paper. This is one of the most powerful features of IGPS because it allows the user to define his own transformations for the axes and therefore provides an unlimited supply of paper types. If you want to use this feature, you must define your transformations before you depress user key number 3 and you must then select user-definable paper.

To define your transformations, you must replace lines 4590 and 4600 in the program with your own functions. The transformation for the x axis must appear on line 4590 and must be of the form

```
4590 DEF FNA(X)=
```

where you supply the information to the right of the equal sign. Line 4600 is for the y axis and must be of a similar form

```
4600 DEF FNB(Y)=
```

To illustrate the power of this feature, consider the data appearing in Table 2.3.

TABLE 2-3 SAMPLE DATA

x	3.00	2.70	2.30	1.80	1.60	1.27	1.10
y	.225	.241	.267	.304	.321	.351	.365

Plotting the data on ordinary, linear, graph paper reveals nothing unusual about them.

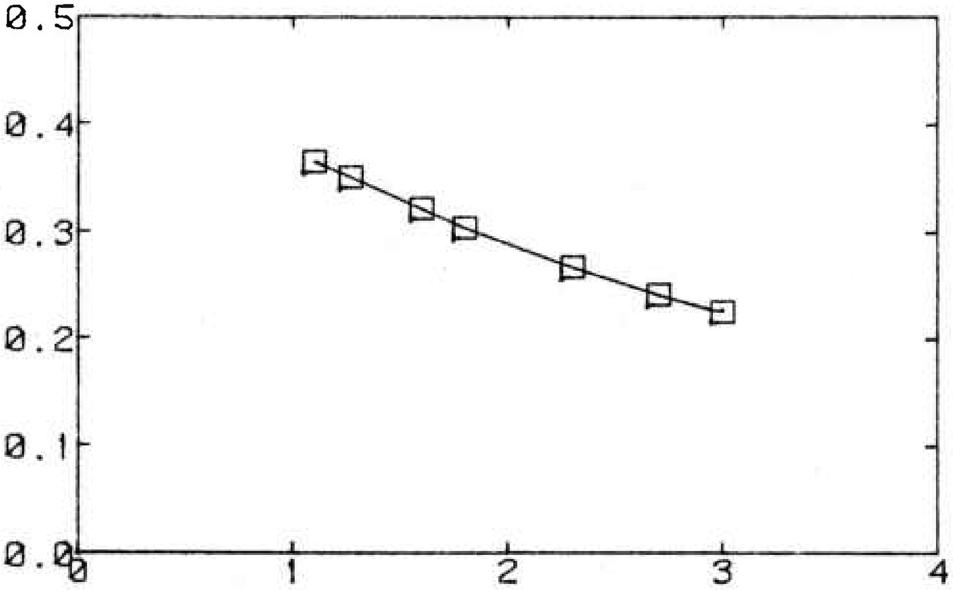

Figure 2.8 Sample Data Plotted on Linear Paper

Incidentally, the source of the data and the units associated with them do not concern us here. What does concern us is that a model of the form

$$y = \frac{k}{\sqrt{x}}$$

has been suggested for the data. To test this model, we type in

```
4590 DEF FNA(X)=-1/SQR(X)

4600 DEF FNB(Y)=Y
```

we select user-definable paper and we ask for a curve fit with a forced intercept of b=0.

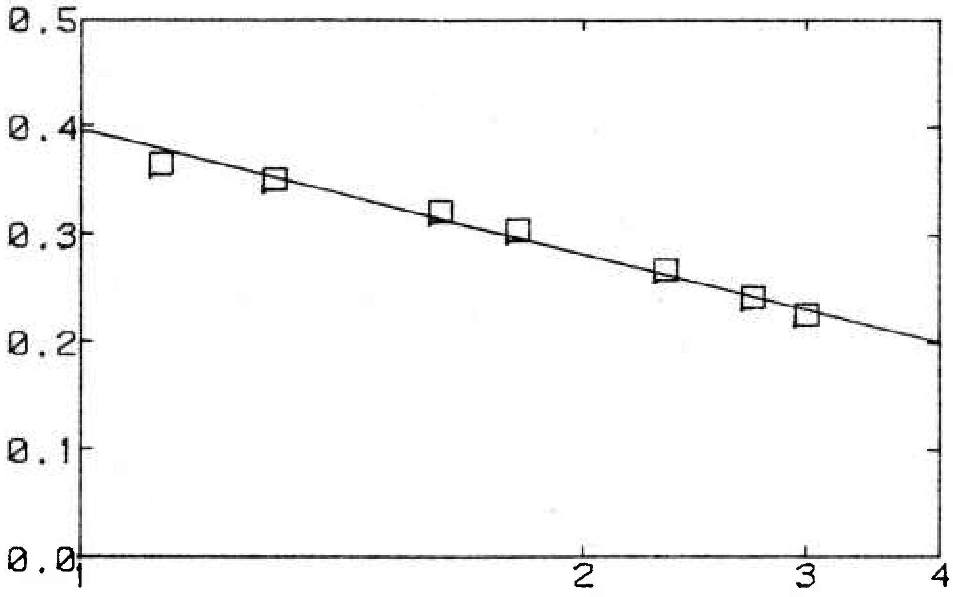

Figure 2.9 Sample Data Plotted on User-Defined Paper

IGPS calculates the slope of the line as a = -0.40. This means that one possible model of the data is:

$$y = \frac{0.40}{\sqrt{x}} \quad .$$

Note that the x axis on user-definable paper starts at 1, whereas the x axis on ordinary paper starts at zero. This difference is necessary because the transformation of zero on user-definable paper would lead to an undefined number (in other words, $\frac{-1}{\sqrt{0}} = -\infty$). Note also that we defined the x-axis transformation as

$$F_a(x) = -\frac{1}{\sqrt{x}} \,, \quad \text{rather than as}$$

$$F_a(x) = \frac{1}{\sqrt{x}} \quad .$$

The minus sign is required to make the transformed value of the minimum along the x axis smaller than the transformed value of the maximum (in other words, if the minus sign had not been included, we would have had $F_a(1) > F_a(4)$, which is not allowed on the Tektronix). The minus sign has to be taken into account when interpreting the results of the curve fitting (which was done in constructing the final model above).

The final point to make about this example is that the value of the free parameter in our model could have been read directly off the graph at the point where the fitted line intersects the vertical line x=1. One way to see this is to note that, in our model, y=k when x=1.

What this example has illustrated is that user-definable paper, in conjunction with curve-fitting, can be used to test the adequacy of a large class of models in representing given data (though no mention of this has yet been made, it will be shown in Section 2.5 that IGPS, when curve-fitting, generates a quantitative esitmate of the goodness-of-fit of the model; this estimate is known as R^2). As another example, consider the data in Table 2-4.

TABLE 2-4 MORE SAMPLE DATA

x	1	2	3	4
y	1.01	2.28	3.92	6.13

One proposed model for these data is:

$$y = \frac{x}{1.1 - kx} .$$

By inverting both sides of the equation and multiplying by -1, this model can be transformed into:

$$-\frac{1}{y} = -\frac{1.1}{x} + k .$$

Therefore, we can check the adequacy of the original model by typing

```
4590 DEF FNA(X)=-1/X
4600 DEF FNB(Y)=-1/Y
```

and then selecting user-defined paper and doing a curve fit with a forced slope of a = 1.1.

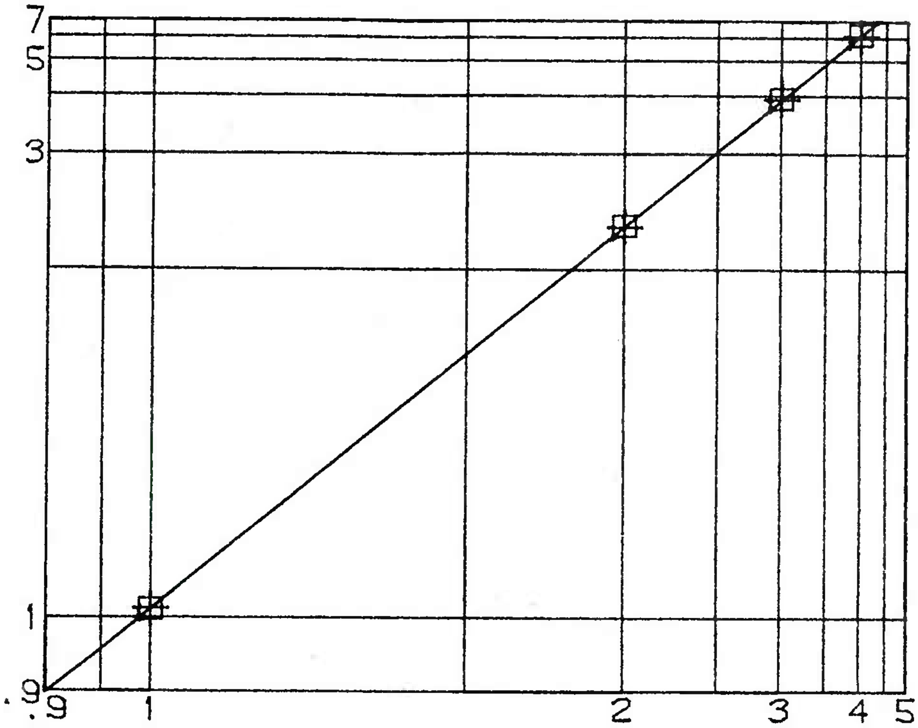

Figure 2.10 Another Example of User-Defined Paper

The plot shows that the model is a good one and as IGPS calculates the intercept of the line as b = 0.11, we know the value of the free parameter k. Therefore our model becomes:

$$y = \frac{x}{1.1 - 0.11x} \quad .$$

This example illustrates that a particular model may need to be manipulated before it can be used to define a paper type. In general, to define a paper type, a model will have to be put into the form:

$$F_b(y) = a\ F_a(x) + b \quad ,$$

where F_a and F_b are functions of x and y and where a and b are functions of the free parameters of the original model.

The paper types described so far have been suitable for sets of two-dimensional, x-versus-y, data. We now describe several types of paper suitable for testing whether a set of data is from a particular probability distribution.

2.3f Uniform Probability Paper. Let us imagine that we have a sample of n points, d_i, i=1,...,n, from a uniform distribution on the interval (A,B). Suppose further that the cumulative fraction of the data associated with the i^{th} point is q_i, i=1,...,n. The q_i's can be determined in two steps: (1) sort the d_i's into increasing order and (2) calculate q_i as:

$$q_i = \frac{i}{1 + n} , \quad i=1,\dots,n .$$

Since the d_i's are uniformly distributed, it must be the case that

$$q_i = F(d_i), \quad i=1,\dots,n ,$$

where F is the cumulative distribution function (c.d.f.) of the uniform distribution on (A,B):

$$q = F(x) = \begin{cases} 0, & x < A, \\ \dfrac{x - A}{B - A}, & A \leq x \leq B, \quad \text{and} \\ 1, & x > B. \end{cases}$$

It follows, therefore, that

$$d_i = F^{-1}(q_i) , \quad i=1,\dots,n ,$$

where F^{-1} represents the inverse of F (the existence of F^{-1} can be guaranteed by restricting the domain of F to the interval $A \leq x \leq B$). In other words,

$$d_i = (B-A) q_i + A, \quad i=1,\dots,n ,$$

so that if we plot the d_i's against the q_i's, we should be able to fit a straight line to the plotted points; what is more, if the slope of the line is a and its intercept is b, we can calculate the parameters of the distribution from a = B - A and b = A; these two equations in fact imply A = b and B = a + b.

Consider the data in Table 2-5.

TABLE 2-5 UNIFORM NUMBERS

5.39	5.87	6.41	5.00
6.76	6.01	6.62	5.64
5.41	6.28	6.87	5.65
5.81	6.50	5.17	6.41
6.83	6.74	5.22	5.16

To check the uniformity of the data, we sort them in increasing order, generate the associated cumulative fractions, and plot them after asking for a curve fit (IGPS will, if requested, do the sorting and the generating of cumulative fractions; note that the fractions are expressed as percentages, however). The plot provides convincing evidence of the uniformity of the numbers; furthermore, the curve fit produces a slope a = 2.24 and an intercept b = 4.87. We conclude that the numbers are from a uniform distribution on (4.87, 7.11).

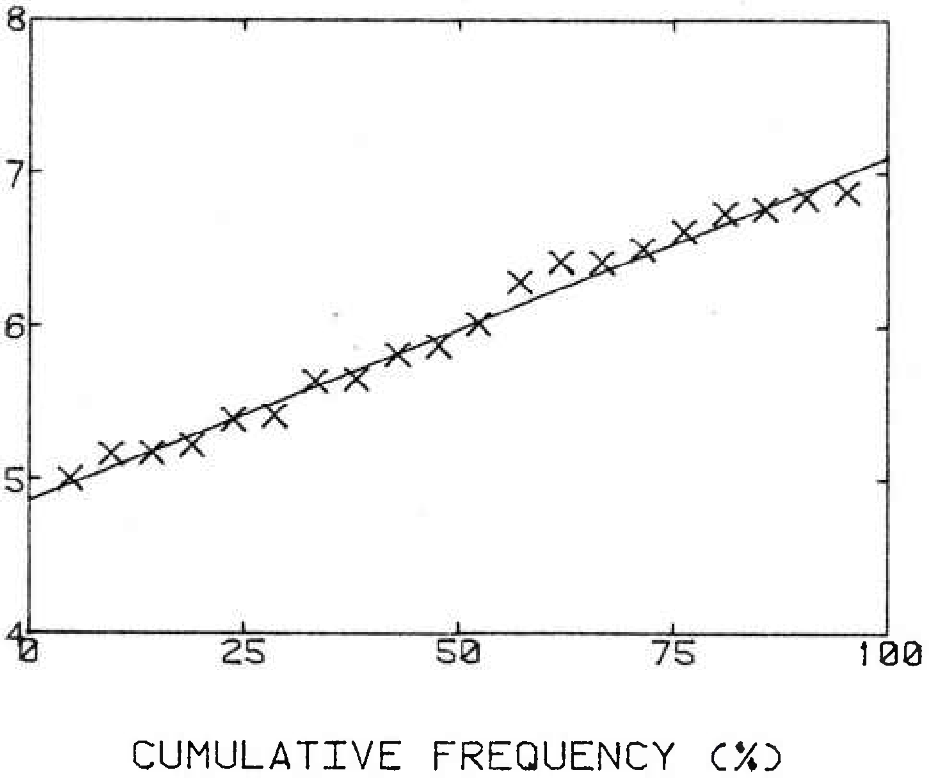

Figure 2.11 Example of the Use of Uniform Probability Paper

2.3g Normal Probability Paper. This paper can be used to check if a given sample of data is from a normal, or gaussian, distribution.

Imagine that we have a sample of n points d_i, i=1,...,n, from a normal distribution with unknown parameters m and s. We can transform the d_i's to standard form by setting

$$z_i = \frac{d_i - m}{s} , \quad i=1,...,n .$$

Furthermore, since the d_i's are normally distributed, then the z_i's are distributed according to the standard normal distribution, and it must therefore be the case that

$$q_i = F(z_i), \quad i=1,...,n ,$$

where the q_i's are the cumulative fractions of the data and F is the c.d.f. of the standard normal distribution:

$$F(x) = \frac{1}{\sqrt{2\Pi}} \int_{-\infty}^{x} e^{-1/2\ t^2}\, dt.$$

It follows that

$$z_i = F^{-1} (q_i), \quad i=1,...,n ,$$

and from the definition of z_i we have

$$\frac{d_i - m}{s} = F^{-1} (q_i), \quad i=1,...,n$$

or

$$d_i = s\ F^{-1} (q_i) + m, \quad i=1,...,n .$$

Therefore, if we plot d_i against $F^{-1}(q_i)$, i=1,...,n, the result will approximate a straight line whose slope and intercept can be used directly to estimate m and s. Consider the data in Table 2-6.

TABLE 2-6 NORMALLY DISTRIBUTED DATA (FREQUENCY)

Frequency	.12	.22	.29	.18	.15	.02
Value	3.0	3.5	4.0	4.5	5.0	5.5

The same data, in cumulative form, appears in Table 2-7.

TABLE 2-7 NORMALLY DISTRIBUTED DATA (CUMULATIVE)

Cumulative Frequency (%)	12	34	63	81	96	98
Value	3.0	3.5	4.0	4.5	5.0	5.5

We enter the data in Table 2-7 into IGPS after selecting normal paper and we do a curve fit.

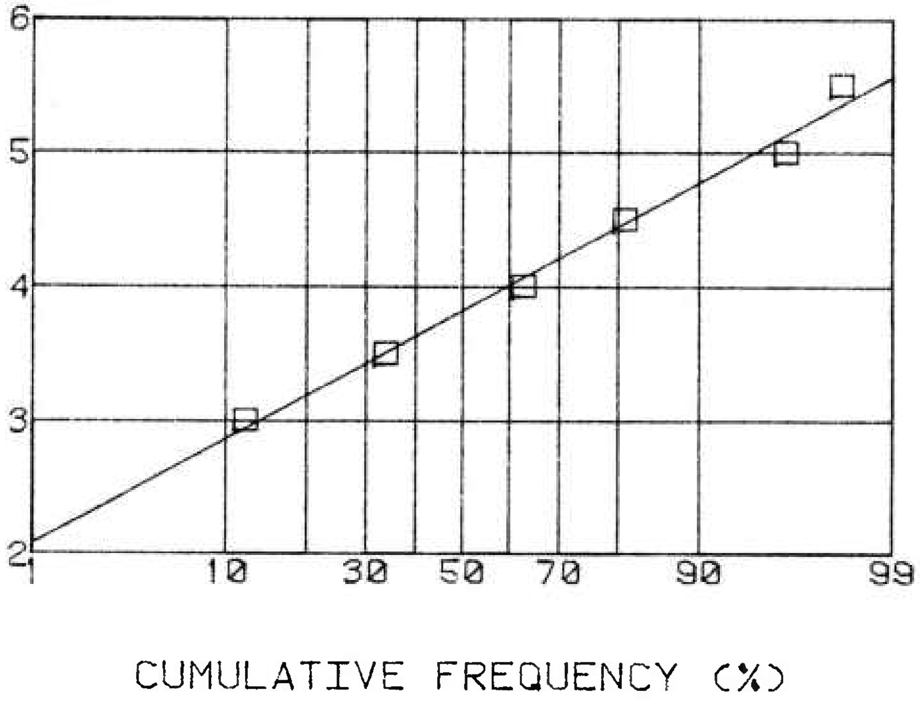

Figure 2.12 Example of the Use of Normal Probability Paper

Estimates of the mean and standard deviation provided by the curve fitting are m = 3.82 and s = 0.75. These two quantities can also be estimated from the graph: the mean is the point at which the line intersects the 50th percentile while the standard deviation is the difference between this value and the value corresponding to a cumulative fraction of 0.8413. These assertions can be verified as follows: our model is of the form

$$d = sF^{-1}(q) + m ,$$

and therefore $d = m$ where $F^{-1}(q) = 0$ and $F^{-1}(q) = 0$ when $q = 0.50$. To find s, note that $s = d - m$ when $F^{-1}(q)=1$ and this is true when $q = 0.8413$. Another way to see this is to note that the mean and median of the standard normal distribution are equal while one standard deviation about the mean spans 68.26 percent of the distribution (and 50 percent plus one-half of 68.26 percent equals 84.13 percent).

Note that the x-axis tick mark labels on the graph avoid the values zero and 100. This should always be done when using normal paper because $F^{-1}(0) = -\infty$ and $F^{-1}(1) = +\infty$.

Finally, a remark about F^{-1} itself: since F^{-1} cannot be expressed in closed form (that is to say in terms of sums and products of functions available on the Tektronix 4051), IGPS uses an approximation of F^{-1} due to Hastings (reference 1, page 192).

The approximation is:

$$F^{-1}(q) = \begin{cases} G(1-q) , & q > .5 , \quad \text{and} \\ -G(q) , & q \leq .5 , \end{cases}$$

where

$$G(q) = z - \frac{h(z)}{s(z)} ,$$

$$z = \ln \frac{1}{q^2} ,$$

$$h(z) = 2.515517 + 0.802853\ z + 0.010328\ z^2 , \text{ and}$$

$$s(z) = 1 + 1.432788\ z + 0.189269\ z^2 + 0.001308\ z^3.$$

It is claimed that the error made in using the approximation is never greater than 0.0004 standard deviations.

2.3h Lognormal Probability Paper. Lognormal paper is similar to normal paper in all respects save one: the y axis of lognormal paper is logrithmic whereas the y axis of normal paper is linear. Therefore, lognormal paper can be used to test whether a sample of data is from a lognormal distribution, that is to say, that the logarithm of the sample is normally distributed.

2.3i Rayleigh Probability Paper. The Rayleigh distribution has c.d.f.

$$q = F(d) = 1 - e^{-\frac{1}{2}\left(\frac{d}{s}\right)^2} .$$

This can be readily transformed into

$$d = s \sqrt{-2 \ \ln \ (1 - q)} \ .$$

Therefore, given a sample of n points d_i, i=1,...,n, with cumulative fractions q_i, i=1,...,n we can test whether the sample is from a Rayleigh distribution by plotting d_i against $\sqrt{-2 \ \ln \ (1 - q_i)}$, i=1,...,n. Furthermore, we can estimate s by fitting a line with intercept b = 0 and slope a to the transformed data and setting s = a. The data in Table 2-8 will illustrate the procedure.

TABLE 2-8 NUMBERS FROM A RAYLEIGH DISTRIBUTION

320	470	16	619
356	547	264	204
429	703	265	305
501	128	468	662
608	143	121	524

We input the data to IGPS, ask IGPS to sort it and generate the cumulative fractions, ask IGPS for a curve fit with a forced intercept of b=0, and plot the results on Rayleigh paper.

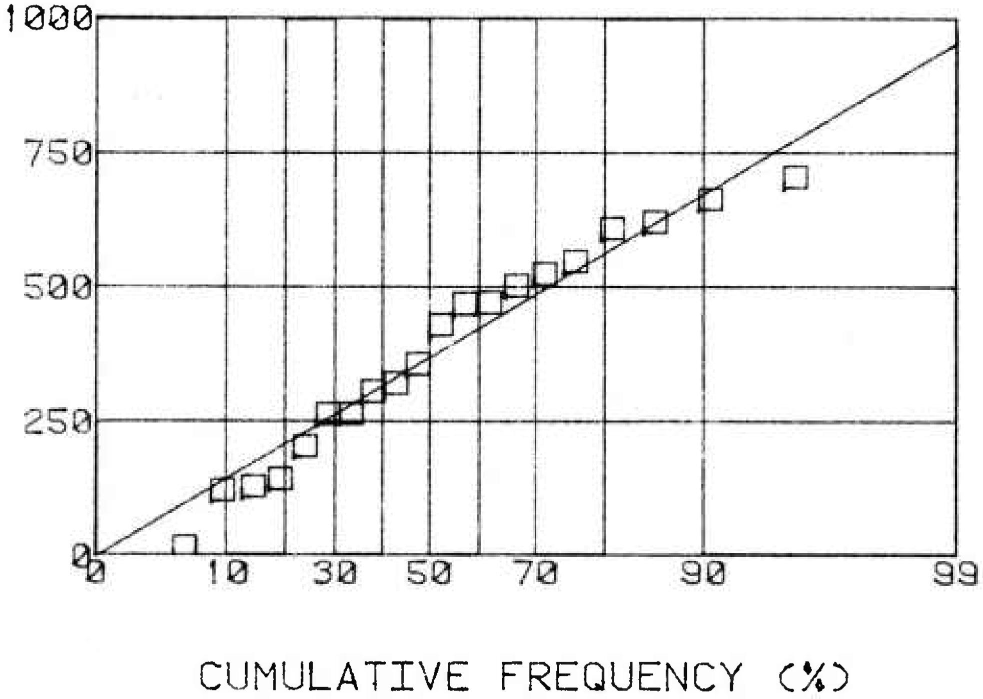

Figure 2.13 Example of the use of Raleigh Probability Paper

One estimate of the parameter s is provided by the curve fit as s = 314. Another is provided directly from the graph at the point where the line intersects the 39th percentile. This is the case because our model is:

$$d = s \sqrt{-2 \ln(1-q)} \ ,$$

and therefore d = s when $\sqrt{-2 \ln (1-q)} = 1$, which is the case when q = 39%.

Note that the value 100 is not used as an x-axis tick mark label; this is because when q = 1, $\sqrt{-2 \ln (1-q)} = +\infty$.

2.3j Weibull Probability Paper. The Weibull distribution has c.d.f.

$$q = F(d) = 1 - e^{-kd^c}, \quad d > 0, \quad k > 0.$$

This can be transformed, by taking logarithms twice, into

$$\ln d = \frac{1}{c} \ln [-\ln (1 - q)] - \frac{1}{c} \ln k.$$

Weibull paper is ruled according to the function

$$F_a(x) = \ln [-\ln (1 - x)]$$

along the x axis and according to the function

$$F_b(y) = \ln y$$

along the y axis. Therefore, given a set of data d_i, i=1,...,n, from a Weibull distribution with parameters c and k, with associated cumulative fractions q_i, i=1,...,n, then, if we plot d_i against q_i, i=1,...,n, on Weibull paper, the result should approximate a straight line. Furthermore, if the slope of the line is a and its intercept is b, then estimates of c and k are given by

$$c = \frac{1}{a} \ , \quad \text{and}$$

$$k = e^{-b/a}$$

Note that, because of the definition of F_a, values of zero and 100 should be avoided on the x-axis (in fact, $F_a(0) = -\infty$ and $F_a(1) = +\infty$); similarly, non-positive values should not be used along the y axis.

Before leaving this section, we should point out that the user can define his own probability paper by using the ideas developed in Section 2.3e. For instance, suppose the user is interested in constructing paper for the exponential distribution. Now the exponential distribution has c.d.f.

$$q = F(d) = 1 - e^{-cd}, \quad c > 0.$$

By rearranging and taking logarithms, this can be converted to

$$d = - \frac{1}{c} \ln (1 - q) .$$

Therefore, to construct exponential paper, the user would type in:

```
4590 DEF FNA(X)=-LOG(1-X/100)

4600 DEF FNB(Y)=Y
```

and would then select user-definable paper. Note that the argument of the function FNA is divided by 100; this is done to allow the x-axis tick marks and labels to be expressed as percentages.

We leave to the reader the problem of determining what kind of curve fit to use with exponential paper and how to estimate the parameter c from the results of the curve fit.

The overall strategy for creating distribution paper, however, is clear: the c.d.f. of the distribution of interest is manipulated into the form

$$F_b(d) = a\ F_a(q) + b,$$

where F_a and F_b are functions of the cumulative frequency of the data and of the data itself and a and b are functions of the free parameters of the distribution.

2.4 User Key Number 4: Input and Output.

This key is used either to input data curves into IGPS or to save previously entered curves from IGPS onto a Tektronix peripheral.

2.4a Input. After the user chooses to input data into IGPS, he will be prompted with

```
DATA TYPE (1 = Y ONLY, 2 = X & Y) :
```

A response of 1 indicates that the user is only going to input y-coordinate data whereas a response of 2 indicates that he is going to input both x-axis and y-axis data. A response of 1 is also a request to IGPS to do two things: (1) sort the y-axis data in increasing order and (2) generate x-axis data as:

$$x_{i,j} = 100 \frac{j}{1 + m_i} \quad , \quad j = 1,\ldots,m_i, \ i = 1,\ldots,n,$$

where n is the number of curves in the data set and m_i is the number of points in the i^{th} curve. This feature is useful with probability paper; see sections 2.3f through 2.3j.

The user is then prompted for the device he wants to read the data curves from. There are four possible responses: one, meaning the Tektronix 4907 File Mangager, a disc system; two, the Tektronix 4924 Auxiliary Tape Drive; three, the Tektronix 4051 Internal Tape Drive; and four, meaning the user wants to input data into IGPS by typing it on the keyboard of the 4051.

The user is then prompted for the number of curves he wants to input and for the number of points in the curve having the largest number of points (in other words, for the number m_0 where $m_0 = \max [\ m_i,\ i = 1,\ldots,n\ ]$, where n is the number of curves being input and m_i is the number of points in the i^{th} curve).

If the user selected the 4907 as the device to use, he is then prompted for the names of the files on which his data reside; if he selected either the 4924 or the internal tape drive, he is then prompted for the numbers of the files; and if he selected the keyboard, he is prompted for the number of points and for the values of the data points in each curve (furthermore, if the user selected y-only data, he is only prompted for the y-axis values, whereas, if he selected x & y data, he is prompted for both x-axis and y-axis values).

When IGPS has read in all the data, it asks for the type of curve marker to be used with each curve. These are nine different curve markers available:

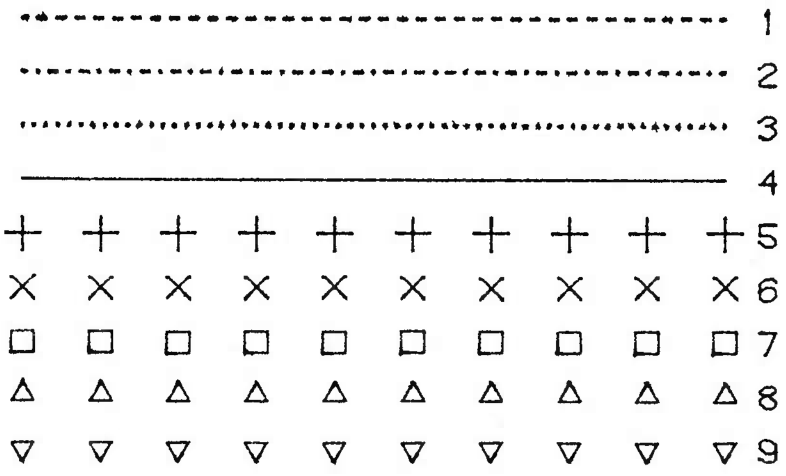

Figure 2.14 Available Curve Markers

Interesting effects can be achieved by plotting the same curve more than once, with a different curve marker each time.

IGPS next prompts for the legend the user wants for each curve; if the user does not want a legend, he should respond with a carriage return. The legend for a particular curve is plotted slightly to the right of the last data point in the curve. This feature was used to create the symbols one through nine in the plot above.

Finally, IGPS prompts for the color the user wants for each curve. Allowable responses are integers between one and eight; these refer to the eight pens in the turret of the Tektronix 4662 x-y plotter equipped with Option 31. The responses are ignored if plotting is either on the screen of the 4051 or on the standard 4662.

If the user selected y-only data, there may now be a slight pause as IGPS sorts the y-axis data and generates the x-axis data. The y-axis data are sorted in place and the original order of the data is therefore lost.

2.4b Output. After the user chooses to output data from IGPS onto a peripheral device, he is prompted for the type of data he wants to save, either y-only or x & y. In the first case, only the y-axis values will be output, whereas in the second, both x-axis and y-axis data are transmitted.

IGPS then prompts for the device the user wishes to output the data to. Possible responses are: one, the 4907; two the 4924; and three, the 4051 Internal Tape. If the user chooses the 4907, IGPS will then ask for the file names on which to write the data; if he chooses the 4924 or the internal tape, it asks for the file numbers.

When writing to the 4907, IGPS will create files of the appropriate size for the data to be written; for tape devices, however, the user must use the MARK command to create the files on his tape before saving data. The tape files must be s bytes long, where

$$s = \begin{cases} 19 + 19 \cdot m \quad , & \text{for y-only data} \\ 19 + 37 \cdot m \quad , & \text{for x \& y data} \end{cases}$$

and m is the number of points in the curve to be written to the file. In practice, the default minimum file size of 768 bytes is adequate for most curves.

If the user attempts to output data without ever having input data during the session, an error message is printed.

2.4c File Structure. Data files to be read by IGPS or written by IGPS are in ASCII format. The structure of an IGPS-compatible file is as follows: (1) the first record in the file contains the number of points in the file; (2) each subsequent record consists of either a single number or of a data pair, depending on whether the file is a y-only or an x & y data file.

2.4d Address Number of the 4924. IGPS assumes that the 4924 Auxiliary Tape Drive is at address 2 of the General Purpose Interface Bus (GPIB). If the user's 4924 is at a different address, he will need to change IGPS. An example will illustrate the required change: if the address wanted is, say, 4 then the user should type in

```
9350 DO = 4  .
```

This must be done before initialization of IGPS.

2.5 User Key Number 5: Curve Fits.

This key is used to do least-squares, linear fits to the user's data. There are several types of fit available: (1) unconstrained, where IGPS calculates both the slope and intercept of the line which best fits the user's data; (2) fixed-slope, where IGPS calculates the intercept of the best line with user-defined slope; and (3) fixed-intercept, where IGPS calculates the slope of the best line with user-defined intercept. A fourth option allows the user to check the goodness of fit of any line he wishes. In all cases, IGPS operates not on the data the user inputs into IGPS but on the data after it has been transformed via the two functions which define the type of paper the user has asked for (see section ˆ.3). For instance, if the user asked for xlog paper, the curve fit is to the logarithm of the x-axis data values.

The following equations are used by IGPS for the different curve fits. Assume we are given a set of transformed data (x_i, y_i), $i = 1,\ldots,n$, where n is the number of points in the set, to which we wish to fit a line of the form $y = ax + b$. The sum of the squares of the vertical deviations between the line and the data points is

$$D = \sum_{i=1}^{n} [y_i - (ax_i + b)]^2 \ .$$

In order to do an unconstrained curve fit, in other words, to calculate both a and b, we differentiate D with respect to a and b and set the results to zero:

$$\frac{dD}{da} = \sum_{i=1}^{n} (2) \left[y_i - (ax_i + b) \right] (-x_i) = 0$$

$$\frac{dD}{db} = \sum_{i=1}^{n} (2) \left[y_i - (ax_i + b) \right] (-1) = 0$$

A little algebra reduces these equations to:

$$a = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i \right)^2} \text{, and}$$

$$b = \frac{1}{n} \left(\sum_{i=1}^{n} y_i - a \sum_{i=1}^{n} x_i \right) .$$

In order to do a fixed-slope curve fit, in other words, in order to calculate b given a, we differentiate D with respect to b and set the result to zero, which yields the equation for b given above. In order to do a fixed-intercept fit, in other words, in order to calculate a given b, we differentiate D with respect to a and set the result to zero, which produces:

$$a = \frac{\sum_{i=1}^{n} x_i y_i - b \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2} .$$

In all cases, IGPS calculates and displays a measure of the goodness of fit of the particular line through the data points. The measure is known as R^2 and is defined as:

$$R^2 = 100 \cdot \left\{ 1 - \frac{D}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \right\} ,$$

where D is as before and $\bar{y}$ is the average of y_i, i=1,...,n:

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \ .$$

From the definition of R^2, it is apparent that the closer R^2 is to 100, the better the fit.

The prompts associated with user key number 5 are straightforward and require no discussion.

2.6 User Key Number 6: 4051.

This key is used to plot the current graph on the screen of the Tektronix 4051. If the graph is not complete, in other words, if some item of information needed to construct the graph has not been supplied to IGPS, the program will prompt the user for the missing item.

This key is particularly useful in previewing a plot: the user can look at the plot on the screen by depressing this key; if he doesn't like what he sees, he can use the other keys (or the shortcuts described in Section 3) to alter the parameters of the plot; finally, he can depress key number 6 again to quickly see the results of any changes he may have made. Of course, if the user takes no action between depressions of the key, then the plot will be re-drawn on the screen without changes.

Once the user is satisfied with the plot, he can get a copy of it on the Tektronix 4662 x-y plotter by depressing user key number 7.

2.7 User Key Number 7: 4662.

This key is used to obtain a hard copy of the current graph on Tektronix 4662 x-y plotter. Before using this key, the user should prepare the 4662 for plotting by placing a piece of paper in the lower left-hand corner of the plotting area of the 4662. If the piece of paper is 11 inches long and eight and one-half inches high, and if the hardware limits of the 4662 are left at their default values, the plot produced by IGPS will be centered on the piece of paper (see Figure 1.1). To produce plots of different sizes, the user can either change the hardware limits of the 4662 or he can modifiy the definition of the viewport in IGPS (see the listing of IGPS in Appendix A at lines 6620 through 6660).

IGPS assumes that the 4662 plotter is at address 1 of the GPIB. If the user's 4662 is at a different address, he will need to change line 28 of the program to read

```
28 D=
```

where the new address goes to the right of the equal sign.

2.8 User Key Number 8: Storage and Retrieval.

This key is used to store onto a peripheral or retrieve from a peripheral all the information required to construct a particular plot. This feature is useful in that it allows a complex plot to be saved during one session with IGPS and then quickly reconstructed at a later session without the user having to re-input all the parameters which make up the plot; instead, the entire plot can be recreated by just depressing this one key and responding to a few prompts.

After the user depresses key number 8, IGPS will prompt him for the type of operation to carry out (either storage or retrieval). The user is then prompted for the device he wishes to use. There are three possible responses: one, meaning the 4907 file manager; two, meaning the 4924 Tape Drive; and three, meaning the internal drive. If the user selected the 4907, he is then prompted for the name of the file to use; if he selected the 4924 or the interval tape drive, he is prompted for the number of the file.

If the user attempts to store an incomplete plot, an error message is printed and no further action is taken.

When storing a plot on the 4907, IGPS will create a binary file of the appropriate size for the data to be written; for tape devices, the user must use the MARK command to create the file on his tape. The tape file must be S bytes long, where S is approximately given by the expression

$$S = 1000 + 16 \cdot n \cdot m_o,$$

where n is the number of data curves in the plot and m_o is the number of points in the curve having the greatest number of points; for instance, storing Figure 1.1 requires approximately 2000 bytes.

The remarks in 2.4d about the address of the 4924 apply here as well.

2.9 User Key Number 9: Data Generation.

This key can be used to generate x axis and y axis data from user-defined parametric functions rather than inputting them from a peripheral or the keyboard.

After the user depresses this key, IGPS will prompt with:

```
DATA TYPE (1=Y ONLY, 2=X&Y):
NUMBER OF CURVES:
NUMBER OF POINTS PER CURVE (AT LEAST 2):
MIN AND MAX OF PARAMETRIC VARIABLE:
```

The meaning of the DATA TYPE prompt is as before: a response of 1 is a request to IGPS to sort the y axis data in increasing order and to generate x axis data as cumulative fractions.

IGPS will, after the user responds to the prompts, create the data as

$$X_{i,j} = F_x \, [\, t_o + S \cdot (j - 1)] \quad \text{and}$$

$$Y_{i,j} = F_y \, [\, t_o + S \cdot (j - 1)] \ ,$$

where $j = 1,\ldots,m_o$, $i=1,\ldots,n$, n and m_o are the responses to the second and third prompts,

$$S = \frac{t_1 - t_o}{m_o - 1} \ ,$$

t_o and t_1 are the minimum and maximum of the parametric variable, and F_x and F_y are the user-definable functions. These two functions must be defined by the user before he depresses user key number 9; he must do so by replacing lines 3420 and 3430 in the program with his own functions: line 3240 is for F_x and must be of the form

```
3420 DEF FNX(T)=
```

while line 3430 is for F_y and must be of the form

```
3430 DEF FNY(T)=
```

By default, the definitions are

```
3420 DEF FNX(T)=RO(I)*SIN(T)

3430 DEF FNY(T)=RO(I)*SIN(T)*COS(T)
```

where RO is an array with five entries such that

$$RO_i = i, \quad i=1,\ldots,5.$$

Therefore, if the user does not alter the definitions, and if he uses the following sequence

```
DATA TYPE (1=Y ONLY, 2=X&Y):  2
NUMBER OF CURVES:  5
NUMBER OF POINTS PER CURVE:  21
MIN AND MAX OF INDEPENDENT VARIABLE:  0, 6.28
```

then, with appropriate choices of limits, labels, curve types and so on, he should be able to reproduce Figure 2.15.

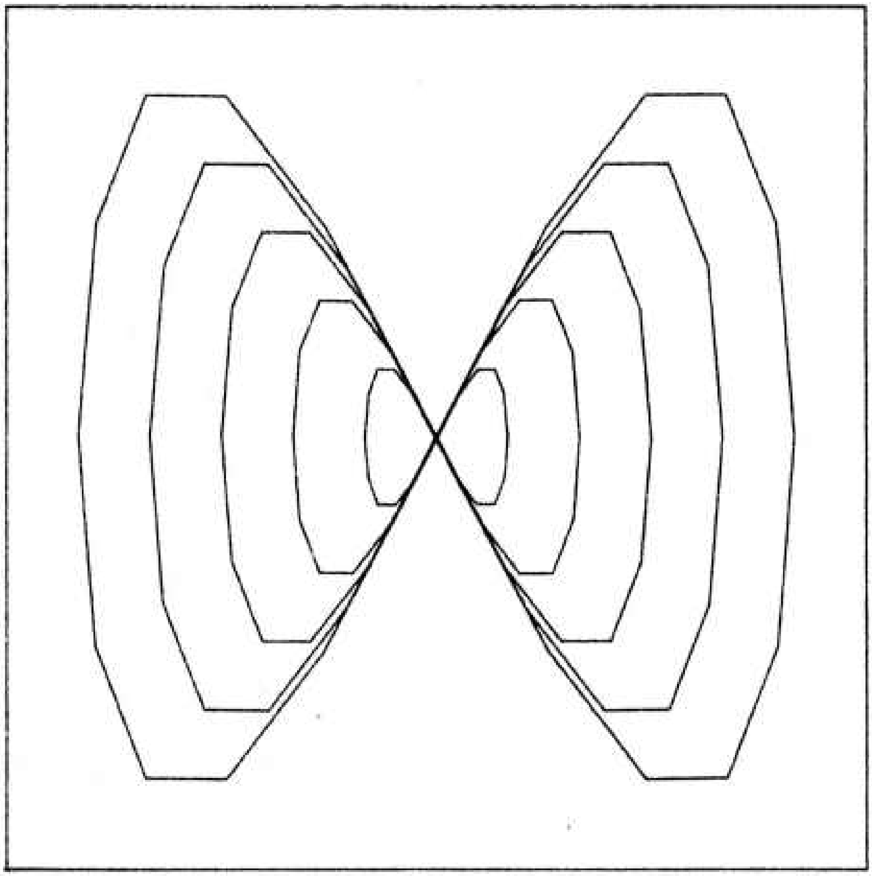

Figure 2.15 Sample Plot Using Parametric Functions

2.10 User Key Number 10: Menu.

Whenever this key is depressed, IGPS will plot the menu on the screen of the 4051; see Figure 2.1.

Next page is blank

3. SHORTCUTS

Suppose you are previewing a plot on the screen of the Tektronix 4051 and you decide to change the type of curve marker being used with one of the curves. One way to do this is to depress user key number 4 and to go through the input procedure; this, however, is clumsy and can be avoided since the variable controlling the type of curve marker is directly available to the user: it is in fact the array LO, with as many entries as there are curves, and with the i^{th} entry of LO being an integer between one and nine (see Figure 2.14) corresponding to the type of curve marker to be used with the i^{th} curve. Therefore, to change the type of curve marker for a given curve, the user need only change the entry in LO corresponding to that curve. For instance, to draw the third curve as a solid line, the user should type in LO(3)=4.

The same procedure can be used with any of the features of the plot being previewed. To find out the name of the variable which controls the feature of the plot which you want to change, consult Table 3-4.

TABLE 3-1 DICTIONARY OF VARIABLES

Name	Type	Description
N1	Simple	Number of entries in X1
X1(N1)	Array	Locations of x axis tick marks
N3	Simple	Number of entries in Y1
Y1(N3)	Array	Locations of y axis tick marks
N2	Simple	Number of entries in A$
A$[a]	String	X axis tick mark labels
N4	Simple	Number of entries in B$
B$[a]	String	Y axis tick mark labels
X$	String	X axis label
Y$	String	Y axis label
Z$	String	Title of plot

[a] see remarks

TABLE 3-1 DICTIONARY OF VARIABLES (CONTINUED)

Name	Type	Description
A0	Simple	Frame type: 0 - None 1 - Axes 2 - Grid 3 - Border
FO	Simple	Color for frame and labels (must be integer between one and eight)
S0	Simple	Pen speed (must be number between 10 and 570)
p[a]	Simple	Paper type: 1 - linear 2 - Xlog 3 - Ylog 4 - Loglog 5 - User-defined 6 - Uniform probability 7 - Normal probability 8 - Lognormal probability 9 - Rayleigh probability 10 - Weibull probability
N	Simple	Number of curves
MO	Simple	Number of points in the curve with the greatest number
M(N)	Array	M(I) contains the number of points in the ith curve
X(N,MO)[a]	Array	X(I,J) contains the jth data point in the ith curve
Y(N,MO)[a]	Array	Y(I,J) contains the jth data point in the ith curve

[a] see remarks

TABLE 3-1 DICTIONARY OF VARIABLES (CONTINUED)

Name	Type	Description
LO(N)	Array	LO(I) contains the curve type for the i^{th} curve: 1 - Dashed line 2 - Dashed-dotted line 3 - Dotted line 4 - Solid line 5 - Crosses 6 - X-shaped markers 7 - Squares 8 - Triangles 9 - Upside-down triangles
F$[a]	String	Legends for the N curves
CO(N)	Array	CO(I) contains the color of the i^{th} curve and legend (must be integer between one and eight)
C	Simple	Curve fit: 0 - No ≠0 - Yes
A(N)[a]	Array	A(I) contains the slope of the line fit to the i^{th} set of data
B(N)[a]	Array	B(I) contains the intercept of the line fit to the i^{th} set of data
R(N)[a]	Array	R(I) contains the R^2 - value associated with the fit to the i^{th} set of data

[a] see remarks

REMARKS:

1. The strings A$ and B$ contain the tick mark labels in compact form, which is to say, A$ consists of N2 labels juxtaposed and separated by pound signs (#). Therefore, the tick mark labels associated with Figure 2.2 could have been created by typing in

```
N2=5

A$="0.00#0.25#0.05#0.75#1.00#"
```

2. Changes to the variable P must be followed by the statement RUN 4410.

3. Unused entries of the arrays X and Y are zero-filled.

4. F$ contains the N legends in compact form (see remark 1 for a definition of compact). The legends in Figure 1.1 could have been created by the statement

```
F$="MARK I# MARK II#"
```

5. The contents of the arrays A, B, and R are only meaningful if a curve-fit has been performed. Otherwise, they are zero-filled.

4. POSTSCRIPT

The author has attempted to make IGPS as useful and versatile as possible and he welcomes recommendations for improving it.

Next page is blank

REFERENCE

1. Hastings, C. Approximations for Digital Computers, Princeton University Press, Princeton, NJ, 1955.

Next page is blank

APPENDIX A
PROGRAM LISTING

Next page is blank

APPENDIX A
PROGRAM LISTING

SET UP USER-DEFINABLE KEYS AND PROGRAM INITIALIZATION
(NOTE: 4662 ASSUMED AT ADDRESS 1 OF GPIB; SEE LINE 28)

```
  1 GOSUB 9330
  2 GO TO 3640
  4 GO TO 100
  8 GO TO 1310
 12 GO TO 4150
 16 GO TO 1410
 20 GO TO 5560
 24 D=32
 25 GO TO 6390
 28 D=1
 29 GO TO 6390
 32 GO TO 4920
 36 GO TO 3370
 40 GO TO 3640
```

USER KEY 1: SET UP AXIS LIMITS AND LABELS

```
100 PAGE
110 DELETE X1,Y1
120 A$=""
130 PRINT "UNIFORM TICKS ON THE X-AXIS (1=YES,2=NO):";
140 INPUT L
```

SELECT TYPE OF X-AXIS LABELING

```
150 GO TO L OF 160,400
```

1.AUTOMATIC X AXIS

```
160 REM
170   M1=0
180   PRINT "XMIN:";
190   GOSUB 1010
200   D1=VAL(W$)
210   PRINT "XMAX:";
220   GOSUB 1010
230   D2=VAL(W$)
240   PRINT "MAJOR TICKS:";
250   GOSUB 1010
260   D4=VAL(W$)
270   PRINT "MINOR TICKS:";
280   INPUT D3
290   N1=1+INT((D2-D1)/D3)
300   DIM X1(N1)
310   FOR I=1 TO N1
320     X1(I)=D1+(I-1)*D3
```

```
330   NEXT I
340   N2=1+INT((D2-D1)/D4)
350   FOR I=D1 TO D2 STEP D4
360     GOSUB 1060
370     A$=A$&W$
380   NEXT I
390   GO TO 540
```

2. USER-CONTROLLED X AXIS

```
400 REM
410   PRINT "NUMBER OF TICK MARKS (AT LEAST 2):";
420   INPUT N1
430   DIM X1(N1)
440   PRINT "LOCATION OF TICK MARKS:";
450   INPUT X1
460   PRINT "NUMBER OF TICK MARK LABELS:";
470   INPUT N2
480   IF N2=0 THEN 530
490     PRINT "LOCATION OF TICK MARK LABELS:";
500     INPUT W$
510     GOSUB 1230
520     A$=W$
530   REM
540 REM
550 B$=""
560 PRINT "UNIFORM TICKS ON THE Y-AXIS (1=YES,2=NO):";
570 INPUT L
```

SELECT TYPE OF Y-AXIS LABELING

```
580 GO TO L OF 590,830
```

1. AUTOMATIC Y AXIS

```
590 REM
600   M1=0
610   PRINT "YMIN:";
620   GOSUB 1010
630   D1=VAL(W$)
640   PRINT "YMAX:";
650   GOSUB 1010
660   D2=VAL(W$)
670   PRINT "MAJOR TICKS:";
680   GOSUB 1010
690   D4=VAL(W$)
700   PRINT "MINOR TICKS:";
710   INPUT D3
720   N3=1+INT((D2-D1)/D3)
730   DIM Y1(N3)
740   FOR I=1 TO N3
750     Y1(I)=D1+(I-1)*D3
760   NEXT I
770   N4=1+INT((D2-D1)/D4)
780   FOR I=D1 TO D2 STEP D4
```

```
 790      GOSUB 1060
 800      B$=B$&W$
 810    NEXT I
 820    GO TO 970
```

2. USER-CONTROLLED Y AXIS

```
 830 REM
 840    PRINT "NUMBER OF TICK MARKS (AT LEAST 2):";
 850    INPUT N3
 860    DIM Y1(N3)
 870    PRINT "LOCATION OF TICK MARKS:";
 880    INPUT Y1
 890    PRINT "NUMBER OF TICK MARK LABELS:";
 900    INPUT N4
 910    IF N4=0 THEN 960
 920      PRINT "LOCATION OF TICK MARK LABELS:";
 930      INPUT W$
 940      GOSUB 1230
 950      B$=W$
 960    REM
 970 REM
 980 K(1)=1
 990 PAGE
1000 RETURN
```

ROUTINE TO CALCULATE THE NUMBER OF DIGITS TO THE RIGHT OF DECIMAL POINT

```
1010 INPUT W$
1020 IF POS(W$,".",1)=0 THEN 1040
1030    M1=M1 MAX LEN(W$)-POS(W$,".",1)
1040 REM
1050 RETURN
```

ROUTINE TO CONSTRUCT TICK MARK LABEL WITH A GIVEN NUMBER PLACES, M1, TO THE RIGHT OF THE DECIMAL POINT

```
1060 W$=STR(ABS(I)*10^M1)
1070 W$=SEG(W$,2,LEN(W$)-1)
1080 IF M1=0 THEN 1170
1090    IF M1<=LEN(W$) THEN 1120
1100      U$=SEG("0000000000",1,M1-LEN(W$))
1110      W$=U$&W$
1120    REM
1130    W$=REP(".",LEN(W$)-M1+1,0)
1140    IF ABS(I)=>1 THEN 1160
1150      W$=REP("0",1,0)
1160    REM
1170 REM
1180 IF I=>0 THEN 1200
1190    W$=REP("-",1,0)
1200 REM
1210 W$=W$&"#"
```

```
1220 RETURN
```

ROUTINE TO REPLACE BLANKS OR COMMAS WITH POUND SIGNS (#)

```
1230 FOR I=1 TO LEN(W$)
1240   U$=SEG(W$,I,1)
1250   IF U$<>" " AND U$<>"," THEN 1270
1260     W$=REP("#",I,1)
1270   REM
1280 NEXT I
1290 W$=W$&"#"
1300 RETURN
```

USER KEY 2: INPUT TITLES

```
1310 PAGE
1320 PRINT "X-AXIS LABEL:";
1330 INPUT X$
1340 PRINT "Y-AXIS LABEL:";
1350 INPUT Y$
1360 PRINT "PLOT TITLE:";
1370 INPUT Z$
1380 K(2)=1
1390 PAGE
1400 RETURN
```

USER KEY 4: INPUT AND OUTPUT OF DATA

```
1410 PAGE
1420 PRINT "INPUT (1) OR SAVE (2):";
1430 INPUT L
1440 PRINT "DATA TYPE (1=Y ONLY,2=X&Y):";
1450 INPUT N0
```

BRANCH ON WHETHER INPUT OR OUTPUT

```
1460 GO TO L OF 1470,2110
1470 REM
1480   PRINT "UNIT NUMBER (1=4907,2=4924,3=INTERNAL TAPE,4=KEYBOARD):";
1490   INPUT L
1500   PRINT "NUMBER OF CURVES:";
1510   INPUT N
1520   PRINT "MAXIMUM NUMBER OF POINTS PER CURVE:";
1530   INPUT M0
1540   DELETE X,Y,M
1550   DIM X(N,M0),Y(N,M0),M(N)
1560   X=0
1570   Y=0
1580   FOR I=1 TO N
```

BRANCH ON DEVICE TYPE

```
1590     GO TO 1*(L=1)+2*(L=2 OR L=3)+3*(L=4) OF 1600,1760,1920
```

1. INPUT FROM 4907 FILE MANAGER

```
1600     REM
1610       PRINT "FILE NAME FOR CURVE NUMBER ";I;":";
1620       INPUT W$
1630       OPEN W$;1,"R",U$
1640       INPUT #1:M(I)
1650       FOR J=1 TO M(I)
1660         GO TO N0 OF 1670,1700
1670         REM
1680           INPUT #1:Y(I,J)
1690           GO TO 1720
1700         REM
1710           INPUT #1:X(I,J),Y(I,J)
1720         REM
1730       NEXT J
1740       CLOSE 1
1750       GO TO 2070
```

2. INPUT FROM 4924 TAPE DRIVE OR INTERNAL DRIVE
(NOTE: 4924 ASSUMED AT ADDRESS 2 OF GPIB; SEE
LINE 9350)

```
1760     REM
1770       PRINT "FILE NUMBER FOR CURVE NUMBER ";I;":";
1780       INPUT D1
1790       D=D0*(L=2)+33*(L=3)
1800       FIND @D:D1
1810       INPUT @D:M(I)
1820       FOR J=1 TO M(I)
1830         GO TO N0 OF 1840,1870
1840         REM
1850           INPUT @D:Y(I,J)
1860           GO TO 1890
1870         REM
1880           INPUT @D:X(I,J),Y(I,J)
1890         REM
1900       NEXT J
1910       GO TO 2070
```

3. INPUT FROM KEYBOARD

```
1920     REM
1930       PAGE
1940       PRINT "NUMBER OF POINTS IN CURVE NUMBER ";I;":";
1950       INPUT M(I)
1960       FOR J=1 TO M(I)
1970         GO TO N0 OF 1980,2020
1980         REM
1990           PRINT "Y(";I;",";J;"):";
2000           INPUT Y(I,J)
```

```
2010           GO TO 2050
2020         REM
2030           PRINT "X(";I;",";J;"),Y(";I;",";J;"):";
2040           INPUT X(I,J),Y(I,J)
2050         REM
2060       NEXT J
2070     REM
2080   NEXT I
2090   GOSUB 2650
2100   GO TO 2630
2110 REM
2120   IF K(4)=0 THEN 2570
2130     PRINT "UNIT NUMBER (1=4907,2=4924,3=INTERNAL TAPE):";
2140     INPUT L
2150     FOR I=1 TO N

                  BRANCH ON DEVICE TYPE

2160       GO TO 1*(L=1)+2*(L=2 OR L=3) OF 2170,2350

                      1. OUTPUT TO 4907 FILE MANAGER

2170       REM
2180         PRINT "FILE NAME FOR CURVE NUMBER ";I;":";
2190         INPUT W$
2200         KILL W$
2210         CREATE W$,"A";19+(19*(N0=1)+37*(N0=2))*M(I),0
2220         OPEN W$;1,"F",U$
2230         PRINT #1:M(I)
2240         FOR J=1 TO M(I)
2250           GO TO N0 OF 2260,2290
2260           REM
2270             PRINT #1:Y(I,J)
2280             GO TO 2310
2290           REM
2300             PRINT #1:X(I,J),Y(I,J)
2310           REM
2320         NEXT J
2330         CLOSE 1
2340         GO TO 2530

                      2. OUTPUT TO 4924 TAPE DRIVE OR INTERNAL DRIVE
                         (NOTE:  4924 ASSUMED AT ADDRESS 2 OF GPIB; SEE
                         LINE 9350)

2350       REM
2360         PRINT "FILE NUMBER FOR CURVE NUMBER ";I;":";
2370         INPUT D1
2380         D=D0*(L=2)+33*(L=3)
2390         KILL @D:D1
2400         FIND @D:D1
2410         PRINT @D:M(I)
2420         FOR J=1 TO M(I)
```

```
2430           GO TO N0 OF 2440,2470
2440           REM
2450             PRINT @D:Y(I,J)
2460             GO TO 2490
2470           REM
2480             PRINT @D:X(I,J),Y(I,J)
2490           REM
2500         NEXT J
2510         PRINT @D,2:
2520         CLOSE
2530       REM
2540     NEXT I
2550     PAGE
2560     GO TO 2620
```

ERROR MESSAGE IF OUTPUT ATTEMPTED WITHOUT INPUT

```
2570   REM
2580     PAGE
2590     PRINT @32,21:35,50
2600     PRINT "NO INPUT YET--NOTHING TO SAVE"
2610     HOME
2620   REM
2630 REM
2640 RETURN
```

INPUT CURVE TYPES

```
2650 PAGE
2660 PRINT "CURVE TYPES AVAILABLE:"
2670 PRINT
2680 PRINT " 1.DASHED LINE"
2690 PRINT " 2.DASHED-DOTTED LINE"
2700 PRINT " 3.DOTTED LINE"
2710 PRINT " 4.SOLID LINE"
2720 PRINT " 5.CROSSES "
2730 PRINT " 6.X'S"
2740 PRINT " 7.SQUARES "
2750 PRINT " 8.TRIANGLES "
2760 PRINT " 9.FLASHES"
2770 PRINT
2780 DELETE L0,C0,A,B,R
2790 DIM L0(N),C0(N),A(N),B(N),R(N)
2800 A=0
2810 B=0
2820 C=0
2830 FOR I=1 TO N
2840   PRINT "TYPE FOR CURVE NUMBER ";I;":";
2850   INPUT L0(I)
2860 NEXT I
```

INPUT LEGENDS

```
2870 PAGE
2880 F$=""
```

```
2890 FOR I=1 TO N
2900   PRINT "LEGEND FOR CURVE NUMBER ";I;":";
2910   INPUT W$
2920   F$=F$&W$
2930   F$=F$&"#"
2940 NEXT I
```

INPUT COLORS

```
2950 PAGE
2960 FOR I=1 TO N
2970   PRINT "COLOR FOR CURVE NUMBER ";I;":";
2980   INPUT C0(I)
2990 NEXT I
```

GENERATE X-AXIS DATA IF REQUESTED

```
3000 IF N0=2 THEN 3330
3010   FOR I=1 TO N
3020     FOR J=1 TO M(I)
3030       X(I,J)=100*J/(1+M(I))
3040     NEXT J
3050   NEXT I
```

SORT Y-AXIS DATA IF REQUESTED (USES BUBBLE SORT)

```
3060   FOR I=1 TO N
3070     IF M(I)<=1 THEN 3310
3080       FOR J=2 TO M(I)
3090         IF Y(I,J-1)<=Y(I,J) THEN 3290
3100           D1=Y(I,J)
3110           L=J+1
3120           D2=1
3130           REM
3140             L=L-1
3150             IF L=1 THEN 3240
3160               IF Y(I,L-1)<=D1 THEN 3190
3170                 Y(I,L)=Y(I,L-1)
3180                 GO TO 3220
3190               REM
3200                 Y(I,L)=D1
3210                 D2=0
3220               REM
3230               GO TO 3270
3240             REM
3250               Y(I,1)=D1
3260               D2=0
3270             REM
3280           IF D2 THEN 3130
3290         REM
3300       NEXT J
3310     REM
3320   NEXT I
3330 REM
```

```
3340 K(4)=1
3350 PAGE
3360 RETURN
```

USER KEY 9: GENERATE DATA FROM FUNCTIONS

```
3370 PAGE
3380 DIM R0(5)
3390 RESTORE 3400
3400 DATA 1,2,3,4,5
3410 READ R0
```

DEFINE FUNCTIONS

```
3420 DEF FNX(T)=R0(I)*SIN(T)
3430 DEF FNY(T)=R0(I)*SIN(T)*COS(T)
3440 PRINT "DATA TYPE (1=Y ONLY,2=X&Y):";
3450 INPUT N0
3460 PRINT "NUMBER OF CURVES:";
3470 INPUT N
3480 PRINT "NUMBER OF POINTS PER CURVE (AT LEAST 2):";
3490 INPUT M0
3500 DELETE X,Y,M
3510 DIM X(N,M0),Y(N,M0),M(N)
3520 M=M0
3530 PRINT "MIN AND MAX OF PARAMETRIC VARIABLE:";
3540 INPUT T0,T1
3550 S=(T1-T0)/(M0-1)
```

GENERATE DATA (NOTE: I INDEXES THE CURVES, J THE POINTS)

```
3560 FOR I=1 TO N
3570   FOR J=1 TO M0
3580     X(I,J)=FNX(T0+S*(J-1))
3590     Y(I,J)=FNY(T0+S*(J-1))
3600   NEXT J
3610 NEXT I
3620 GOSUB 2650
3630 RETURN
```

USER KEY 10: DRAW MENU

```
3640 D=32
3650 PAGE @D:
3660 DIM K$(80)
3670 K$=" LIMITS  LABELS  FORMAT    IO     FITS  "
3680 K$=K$&"  4051    4662     S&R  FUNCTION  MENU  "
3690 L=60
3700 D2=L
3710 FOR J=1 TO 2
3720   D1=-4
3730   FOR I=1 TO 5
```

```
3740     D1=D1+20
3750     PRINT @D,21:D1,D2
3760     PRINT @D,20:D1+16,D2,D1+16,D2-4,D1,D2-4,D1,D2
3770   NEXT I
3780   D2=D2-17.6
3790 NEXT J
3800 PRINT @D,21:23.8,L-8.4
3810 PRINT @D:"1"
3820 PRINT @D,21:43.8,L-8.4
3830 PRINT @D:"2"
3840 PRINT @D,21:63.8,L-8.4
3850 PRINT @D:"3"
3860 PRINT @D,21:83.8,L-8.4
3870 PRINT @D:"4"
3880 PRINT @D,21:103.8,L-8.4
3890 PRINT @D:"5"
3900 PRINT @D,21:16,L-10.8
3910 PRINT @D,20:50,L-10.8
3920 PRINT @D,21:52,L-12
3930 PRINT @D:"USER DEFINABLE"
3940 PRINT @D,21:78,L-10.8
3950 PRINT @D,20:112,L-10.8
3960 PRINT @D,21:23.8,L-26
3970 PRINT @D:"6"
3980 PRINT @D,21:43.8,L-26
3990 PRINT @D:"7"
4000 PRINT @D,21:63.8,L-26
4010 PRINT @D:"8"
4020 PRINT @D,21:83.8,L-26
4030 PRINT @D:"9"
4040 PRINT @D,21:102.5,L-26
4050 PRINT @D:"10"
4060 FOR J=1 TO 2
4070   FOR I=1 TO 5
4080     PRINT @D,21:17+20*(I-1),L-3.2-17.6*(J-1)
4090     U$=SEG(K$,(J-1)*40+(I-1)*8+1,8)
4100     PRINT @D:U$
4110   NEXT I
4120 NEXT J
4130 HOME @D:
4140 RETURN
```

USER KEY 3: FORMAT DEFINITION

```
4150 PAGE
4160 PRINT "FRAME TYPE (0=NONE,1=AXES,2=GRID,3=BORDER):";
4170 INPUT A0
4180 PRINT "COLOR FOR FRAME AND LABELS (1-8):";
4190 INPUT F0
4200 PRINT "PEN SPEED (10-570 MM/S):";
4210 INPUT S0
4220 PRINT
4230 PRINT "   PAPER TYPES:               X-AXIS      Y-AXIS"
```

```
4240 PRINT
4250 PRINT " 1.     LINEAR:                    X           Y"
4260 PRINT " 2.       XLOG:               LOG(X)           Y"
4270 PRINT " 3.       YLOG:                    X      LOG(Y)"
4280 PRINT " 4.     LOGLOG:               LOG(X)      LOG(Y)"
4290 PRINT " 5.       USER:                    ?           ?"
4300 PRINT " 6.    UNIFORM:                X/100           Y"
4310 PRINT " 7.     NORMAL:             N(X/100)           Y"
4320 PRINT " 8.  LOGNORMAL:             N(X/100)      LOG(Y)"
4330 PRINT " 9.   RAYLEIGH: SQR(-2*LOG(1-X/100))           Y"
4340 PRINT "10.    WEIBULL:   LOG(-LOG(1-X/100))      LOG(Y)"
4350 PRINT
4360 PRINT "PAPER TYPE:";
4370 INPUT P
4380 GOSUB 4410
4390 K(3)=1
4400 RETURN
```

BRANCH ON PAPER TYPE SELECTED FOR FUNCTION DEFINITION

```
4410 GO TO P OF 4420,4460,4500,4540,4580,4620,4660,4740,4820,4860
4420 REM
4430   DEF FNA(X)=X
4440   DEF FNB(Y)=Y
4450   GO TO 4890
4460 REM
4470   DEF FNA(X)=LOG(X)
4480   DEF FNB(Y)=Y
4490   GO TO 4890
4500 REM
4510   DEF FNA(X)=X
4520   DEF FNB(Y)=LOG(Y)
4530   GO TO 4890
4540 REM
4550   DEF FNA(X)=LOG(X)
4560   DEF FNB(Y)=LOG(Y)
4570   GO TO 4890
4580 REM
4590   DEF FNA(X)=X
4600   DEF FNB(Y)=Y
4610   GO TO 4890
4620 REM
4630   DEF FNA(X)=X/100
4640   DEF FNB(Y)=Y
4650   GO TO 4890
4660 REM
4670   DEF FNA(X)=FNC((1-X/100)*(X>50)+X/100*(X<=50))*((X>50)-(X<=50))
4680   DEF FNC(X)=FND(X)-FNE(FND(X))/FNF(FND(X))
4690   DEF FND(X)=SQR(LOG(1/(X*X)))
4700   DEF FNE(X)=2.515517+(0.802853+0.010328*X)*X
4710   DEF FNF(X)=1+(1.432788+(0.189269+0.001308*X)*X)*X
4720   DEF FNB(Y)=Y
4730   GO TO 4890
4740 REM
```

```
4750   DEF FNA(X)=FNC((1-X/100)*(X>50)+X/100*(X<=50))*((X>50)-(X<=50))
4760   DEF FNC(X)=FND(X)-FNE(FND(X))/FNF(FND(X))
4770   DEF FND(X)=SQR(LOG(1/(X*X)))
4780   DEF FNE(X)=2.515517+(0.802853+0.010328*X)*X
4790   DEF FNF(X)=1+(1.432788+(0.189269+0.001308*X)*X)*X
4800   DEF FNB(Y)=LOG(Y)
4810   GO TO 4890
4820 REM
4830   DEF FNA(X)=SQR(-2*LOG(1-X/100))
4840   DEF FNB(Y)=Y
4850   GO TO 4890
4860 REM
4870   DEF FNA(X)=LOG(-LOG(1-X/100))
4880   DEF FNB(Y)=LOG(Y)
4890 REM
4900 PAGE
4910 RETURN
```

USER KEY 8: STORE AND RETRIEVE PLOTS

```
4920 PAGE
4930 PRINT "STORE (1) OR RETRIEVE (2):";
4940 INPUT L
4950 PRINT "UNIT NUMBER (1=4907,2=4924,3=INTERNAL TAPE):";
4960 INPUT D1
```

BRANCH ON STORAGE OR RETRIEVAL

```
4970 GO TO L OF 4980,5310
4980 REM
4990   IF SUM(K)<>5 THEN 5240
```

BRANCH ON DEVICE TYPE

```
5000     GO TO 1*(D1=1)+2*(D1=2 OR D1=3) OF 5010,5110
```

1. STORAGE ON 4907 FILE MANAGER

```
5010     REM
5020       PRINT "FILE NAME:";
5030       INPUT W$
5040       KILL W$
5050       CREATE W$;845+8*(N1+N3+N*(2*M0+5)),0
5060       OPEN W$;1,"F",U$
5070       WRITE #1:N1,N2,N3,N4,A$,B$,X$,Y$,Z$,N,M0,F$,A0,F0,S0,P,C
5080       WRITE #1:X1,Y1,X,Y,M,L0,C0,A,B
5090       CLOSE 1
5100       GO TO 5210
```

2. STORAGE ON 4924 TAPE DRIVE OR INTERNAL DRIVE (NOTE: 4924 ASSUMED AT ADDRESS 2 OF GPIB; SEE LINE 9350)

```
5110      REM
5120        PRINT "FILE NUMBER:";
5130        INPUT D2
5140        D=D0*(D1=2)+33*(D1=3)
5150        KILL @D:D2
5160        FIND @D:D2
5170        WRITE @D:N1,N2,N3,N4,A$,B$,X$,Y$,Z$,N,M0,F$,A0,F0,S0,P,C
5180        WRITE @D:X1,Y1,X,Y,M,L0,C0,A,B
5190        PRINT @D,2:
5200        CLOSE
5210      REM
5220      PAGE
5230      GO TO 5290
```

ERROR MESSAGE IF STORAGE ATTEMPTED BEFORE PLOT IS READY

```
5240    REM
5250      PAGE
5260      PRINT @32,21:30,50
5270      PRINT "PLOT NOT COMPLETE--NOT STOREABLE"
5280      HOME
5290    REM
5300    GO TO 5540
5310  REM
5320    DELETE X1,Y1,X,Y,M,L0,C0,A,B,R
```

BRANCH ON DEVICE TYPE

```
5330    GO TO 1*(D1=1)+2*(D1=2 OR D1=3) OF 5340,5430
```

1. RETRIEVAL FROM 4907 FILE MANAGER

```
5340    REM
5350      PRINT "FILE NAME:";
5360      INPUT W$
5370      OPEN W$;1,"R",U$
5380      READ #1:N1,N2,N3,N4,A$,B$,X$,Y$,Z$,N,M0,F$,A0,F0,S0,P,C
5390      DIM X1(N1),Y1(N3),X(N,M0),Y(N,M0),M(N),L0(N),C0(N),A(N),B(N),R(
5400      READ #1:X1,Y1,X,Y,M,L0,C0,A,B
5410      CLOSE 1
5420      GO TO 5510
```

2. RETRIEVAL FROM 4924 TAPE DRIVE OR INTERNAL DRIVE (NOTE: 4924 ASSUMED AT ADDRESS 2 OF GPIB; SEE LINE 9350)

```
5430    REM
5440      PRINT "FILE NUMBER:";
5450      INPUT D2
5460      D=D0*(D1=2)+33*(D1=3)
```

```
5470       FIND @D:D2
5480       READ @D:N1,N2,N3,N4,A$,B$,X$,Y$,Z$,N,M0,F$,A0,F0,S0,P,C
5490       DIM X1(N1),Y1(N3),X(N,M0),Y(N,M0),M(N),L0(N),C0(N),A(N),B(N),R(N)
5500       READ @D:X1,Y1,X,Y,M,L0,C0,A,B
5510     REM
5520     GOSUB 4410
5530     K=1
5540 REM
5550 RETURN
```

USER KEY 5: CURVE FITS

```
5560 PAGE
5570 IF K(4)=0 THEN 6330
5580   PRINT "CURVE FITS (0=NO,1=YES):";
5590   INPUT C
5600   IF C=0 THEN 6280
5610     IF K(3) THEN 5630
5620       GOSUB 4150
5630     REM
5640     PRINT
5650     PRINT "CURVE FITS AVAILABLE:"
5660     PRINT
5670     PRINT " 1.UNCONSTRAINED"
5680     PRINT " 2.FIX SLOPE"
5690     PRINT " 3.FIX INTERCEPT"
5700     PRINT " 4.FIX BOTH"
5710     PRINT
5720     PRINT "CURVE FIT DESIRED:";
5730     INPUT C
```

TRANSFORM DATA VIA FUNCTIONS DEFINING PAPER TYPE

```
5740     FOR I=1 TO N
5750       DELETE X0,Y0,Z0
5760       DIM X0(M(I)),Y0(M(I)),Z0(M(I))
5770       FOR J=1 TO M(I)
5780         X0(J)=FNA(X(I,J))
5790         Y0(J)=FNB(Y(I,J))
5800       NEXT J
```

BRANCH ON CURVE FIT SELECTED

```
5810       GO TO C OF 5820,5890,5940,6020
5820       REM
5830         Z0=X0*Y0
5840         A(I)=M(I)*SUM(Z0)-SUM(X0)*SUM(Y0)
5850         Z0=X0*X0
5860         A(I)=A(I)/(M(I)*SUM(Z0)-SUM(X0)^2)
5870         B(I)=(SUM(Y0)-A(I)*SUM(X0))/M(I)
5880         GO TO 6050
5890       REM
5900         PRINT "SLOPE OF CURVE NUMBER ";I;":";
```

```
5910         INPUT A(I)
5920         B(I)=(SUM(Y0)-A(I)*SUM(X0))/M(I)
5930         GO TO 6050
5940       REM
5950         PRINT "INTERCEPT OF CURVE NUMBER ";I;":";
5960         INPUT B(I)
5970         Z0=X0*Y0
5980         A(I)=SUM(Z0)-B(I)*SUM(X0)
5990         Z0=X0*X0
6000         A(I)=A(I)/SUM(Z0)
6010         GO TO 6050
6020       REM
6030         PRINT "SLOPE, INTERCEPT OF CURVE NUMBER ";I;":";
6040         INPUT A(I),B(I)
6050       REM
```

CALCULATE R-SQUARE

```
6060       Z0=SUM(Y0)/M(I)
6070       Z0=Y0-Z0
6080       Z0=Z0*Z0
6090       R(I)=1/SUM(Z0)
6100       Z0=A(I)*X0
6110       Z0=B(I)+Z0
6120       Z0=Y0-Z0
6130       Z0=Z0*Z0
6140       R(I)=100*(1-SUM(Z0)*R(I))
6150     NEXT I
```

PRINT OUT RESULTS OF CURVE FITS

```
6160     PAGE
6170     FOR I=1 TO (34-(N+3))/2
6180       PRINT
6190     NEXT I
6200     PRI USI 6210:"   CURVE #","     SLOPE"," INTERCEPT","  R-SQUARE"
6210     IMAGE 8X,4(10A,5X)
6220     PRINT
6230     FOR I=1 TO N
6240       PRINT USING "8X,5X,2D,3X,3(5X,7D.2D)":I,A(I),B(I),R(I)
6250     NEXT I
6260     HOME
6270     GO TO 6300
6280   REM
6290     PAGE
6300   REM
6310   K(5)=1
6320   GO TO 6370
```

ERROR MESSAGE IF FIT ATTEMPTED BEFORE DATA INPUT

```
6330 REM
6340   PRINT @32,21:25,50
6350   PRINT "NO DATA INPUT YET--CURVE FIT IMPOSSIBLE"
```

```
6360   HOME
6370 REM
6380 RETURN
```

USER KEYS 6 AND 7: PLOT ON 4051 OR 4662

```
6390 PAGE @D:
```

CHECK FOR MISSING PLOT PARAMETERS

```
6400 IF K(1) THEN 6420
6410   GOSUB 100
6420 REM
6430 IF K(2) THEN 6450
6440   GOSUB 1310
6450 REM
6460 IF K(3) THEN 6480
6470   GOSUB 4150
6480 REM
6490 IF K(4) THEN 6510
6500   GOSUB 1410
6510 REM
6520 IF K(5) THEN 6540
6530   GOSUB 5560
6540 REM
```

SELECT PEN COLOR AND SPEED

```
6550 PRINT @D,8:F0
6560 PRINT @D,32:"BY";S0
```

DEFINE WINDOW IN UDU'S

```
6570 U1=FNA(X1(1))
6580 U2=FNA(X1(N1))
6590 U3=FNB(Y1(1))
6600 U4=FNB(Y1(N3))
6610 WINDOW U1,U2,U3,U4
```

DEFINE VIEWPORT IN GDU'S

D1 = HORIZONTAL LOCATION OF LOWER-LEFT CORNER
D2 = HORIZONTAL EXTENT
D3 = VERTICAL LOCATION OF LOWER-LEFT CORNER
D4 = VERTICAL EXTENT

```
6620 D1=26*(D<>32)+40*(D=32)
6630 D2=45*(D<>32)+60*(D=32)
6640 D3=15*(D<>32)+20*(D=32)
6650 D4=45*(D<>32)+60*(D=32)
6660 VIEWPORT D1,D1+D2,D3,D3+D4
```

CALCULATE SCALES AND TICK LENGTHS

```
6670 X2=(U2-U1)/D2
6680 Y2=(U4-U3)/D4
6690 X3=X2*(D2 MAX D4)/75
6700 Y3=Y2*(D2 MAX D4)/75
```

DEFINE DEFAULT HARDWARE CHARACTER SIZES

```
6710 C1=1.792*(D<>32)+1.79*(D=32)
6720 C2=1.195*(D<>32)+1.195*(D=32)
6730 C3=2.816*(D<>32)+2.6*(D=32)
6740 C4=1.721*(D<>32)+2.2*(D=32)
```

DEFINE RELATIVE CHARACTER SIZES FOR TICK MARK LABELS, AXIS LABELS, PLOT TILE, AND LEGENDS

```
6750 S1=0.8*(D<>32)+1*(D=32)
6760 S2=1*(D<>32)+1*(D=32)
6770 S3=0.7*(D<>32)+1*(D=32)
```

DEFINE OFFSETS FOR AXIS LABELS AND PLOT TITLE

```
6780 F1=7*(D<>32)+10*(D=32)
6790 F2=10*(D<>32)+14*(D=32)
6800 F3=7*(D<>32)+9*(D=32)
```

BRANCH ON FRAME TYPE (ZERO MEANS NO FRAME OR LABELS)

```
6810 IF A0=0 THEN 8060
6820   GO TO A0 OF 6830,6990,7190
```

1. DRAW AXES

```
6830   REM
6840     MOVE @D:U1,U4
6850     FOR I=N3 TO 2 STEP -1
6860       D2=FNB(Y1(I))
6870       DRAW @D:U1,D2
6880       DRAW @D:U1+X3,D2
6890       DRAW @D:U1,D2
6900     NEXT I
6910     DRAW @D:U1,U3
6920     FOR I=2 TO N1
6930       D1=FNA(X1(I))
6940       DRAW @D:D1,U3
6950       DRAW @D:D1,U3+Y3
6960       DRAW @D:D1,U3
6970     NEXT I
6980     GO TO 7570
```

2. DRAW GRID

```
6990    REM
7000      MOVE @D:Ul,U3
7010      DRAW @D:Ul,U4
7020      D2=U4
7030      FOR I=2 TO Nl
7040        Dl=FNA(Xl(I))
7050        DRAW @D:Dl,D2
7060        D2=U3*(D2=U4)+U4*(D2=U3)
7070        DRAW @D:Dl,D2
7080      NEXT I
7090      DRAW @D:U2,U3
7100      DRAW @D:Ul,U3
7110      Dl=Ul
7120      FOR I=2 TO N3
7130        D2=FNB(Yl(I))
7140        DRAW @D:Dl,D2
7150        Dl=Ul*(Dl=U2)+U2*(Dl=Ul)
7160        DRAW @D:Dl,D2
7170      NEXT I
7180      GO TO 7570
```

3. DRAW BORDER

```
7190    REM
7200      MOVE @D:Ul,U3
7210      IF Nl=2 THEN 7280
7220        FOR I=2 TO Nl-l
7230          Dl=FNA(Xl(I))
7240          DRAW @D:Dl,U3
7250          DRAW @D:Dl,U3+Y3
7260          DRAW @D:Dl,U3
7270        NEXT I
7280      REM
7290      DRAW @D:U2,U3
7300      IF N3=2 THEN 7370
7310        FOR I=2 TO N3-l
7320          D2=FNB(Yl(I))
7330          DRAW @D:U2,D2
7340          DRAW @D:U2-X3,D2
7350          DRAW @D:U2,D2
7360        NEXT I
7370      REM
7380      DRAW @D:U2,U4
7390      IF Nl=2 THEN 7460
7400        FOR I=Nl-l TO 2 STEP -l
7410          Dl=FNA(Xl(I))
7420          DRAW @D:Dl,U4
7430          DRAW @D:Dl,U4-Y3
7440          DRAW @D:Dl,U4
7450        NEXT I
7460      REM
7470      DRAW @D:Ul,U4
```

```
7480     IF N3=2 THEN 7550
7490      FOR I=N3-1 TO 2 STEP -1
7500       D2=FNB(Y1(I))
7510       DRAW @D:U1,D2
7520       DRAW @D:U1+X3,D2
7530       DRAW @D:U1,D2
7540      NEXT I
7550     REM
7560     DRAW @D:U1,U3
7570    REM
```

PRINT X-AXIS LABEL

```
7580    PRINT @D,17:S1*C1,S1*C3
7590    PRINT @D,25:0
7600    MOVE @D:(U1+U2-S1*X2*(C1*(LEN(X$)-1)+C2))/2,U3-F1*Y2
7610    PRINT @D:X$
```

PRINT Y-AXIS LABEL

```
7620    IF D=32 THEN 7670
7630     PRINT @D,25:90
7640     MOVE @D:U1-(F2-S1*C4)*X2,(U3+U4-S1*Y2*(C1*(LEN(Y$)-1)+C2))/2
7650     PRINT @D:Y$
7660     GO TO 7730
7670    REM
7680     FOR I=1 TO LEN(Y$)
7690      MOVE U1-F2*X2,(U3+U4+Y2*S1*(C3*(LEN(Y$)-1)+C4))/2-Y2*(C4+C3*(I-1))
7700      W$=SEG(Y$,I,1)
7710      PRINT W$
7720     NEXT I
7730    REM
```

PRINT AND UNDERLINE PLOT TITLE

```
7740    IF LEN(Z$)=0 THEN 7830
7750     PRINT @D,17:S2*C1,S2*C3
7760     PRINT @D,25:0
7770     MOVE @D:(U1+U2-S2*X2*(C1*(LEN(Z$)-1)+C2))/2,U4+F3*Y2
7780     PRINT @D:Z$
7790     MOVE @D,21:(U1+U2+S2*X2*(C1*(LEN(Z$)-1)+C2))/2,U4+(F3-0.2*S2*C3)*Y2
7800     MOVE @D,20:(U1+U2-S2*X2*(C1*(LEN(Z$)-1)+C2))/2,U4+(F3-0.2*S2*C3)*Y2
7810     MOVE @D,21:(U1+U2-S2*X2*(C1*(LEN(Z$)-1)+C2))/2,U4+(F3-0.4*S2*C3)*Y2
7820     MOVE @D,20:(U1+U2+S2*X2*(C1*(LEN(Z$)-1)+C2))/2,U4+(F3-0.4*S2*C3)*Y2
7830    REM
7840    PRINT @D,17:S3*C1,S3*C3
7850    PRINT @D,25:0
```

PRINT TICK MARK LABELS ON X AXIS

```
7860    IF N2=0 THEN 7950
7870     U$=A$
7880     FOR I=1 TO N2
7890      W$=SEG(U$,1,POS(U$,"#",1)-1)
```

```
7900        U$=SEG(U$,POS(U$,"#",1)+1,LEN(U$)-POS(U$,"#",1))
7910        D1=FNA(VAL(W$))
7920        MOVE @D:D1-S3*X2*(C1*(LEN(W$)-1)+C2)/2,U3-S3*Y2*(1.25*C4)
7930        PRINT @D:W$
7940      NEXT I
7950    REM

              PRINT TICK MARK LABELS ON Y AXIS

7960    IF N4=0 THEN 8050
7970      U$=B$
7980      FOR I=1 TO N4
7990        W$=SEG(U$,1,POS(U$,"#",1)-1)
8000        U$=SEG(U$,POS(U$,"#",1)+1,LEN(U$)-POS(U$,"#",1))
8010        D1=FNB(VAL(W$))
8020        MOVE @D:U1-S3*X2*(C1*LEN(W$)-0.2*(C1-C2)),D1-S3*Y2*(1.125*C4)/2
8030        PRINT @D:W$
8040      NEXT I
8050    REM
8060 REM
8070 PRINT @D,17:S3*C1,S3*C3
8080 PRINT @D,25:0
8090 U$=F$

              LOOP OVER N DATA CURVES

8100 FOR I=1 TO N

              1. SELECT COLOR FOR I'TH CURVE

8110    PRINT @D,8:C0(I)

              2. BRANCH ON CURVE TYPE

8120    GO TO 1*(L0(I)<=3)+2*(L0(I)=4)+3*(L0(I)=>5) OF 8130,8710,8810

                2.A DASHED LINES (CURVE TYPE 1 THROUGH 3)

8130    REM
8140      IF M(I)<=1 THEN 8690

                  2.A.1 LOAD IN DASHED LINE PARAMETERS (IN GDU'S)
                        (NOTE: 1'ST DASHED LINE HAS 3 ELEMENTS,
                        2'ND HAS 5,3'RD HAS 3)

8150        RESTORE 8160
8160        DATA 3,0.2,0.6,0.2,5,0.2,0.6,0.01,0.6,0.2,3,0.005,0.6,0.005
8170        FOR J=1 TO L0(I)
8180          READ L
8190          DELETE P0
8200          DIM P0(L)
8210          READ P0
8220        NEXT J
```

2.A.2 TRANSFORM DATA

```
8230        DELETE X0,Y0
8240        DIM X0(M(I)),Y0(M(I))
8250        FOR J=1 TO M(I)
8260          X0(J)=FNA(X(I,J))
8270          Y0(J)=FNB(Y(I,J))
8280        NEXT J
```

2.A.3 CALCULATE CURVE LENGTH IN GDU'S

```
8290        T0=0
8300        FOR J=2 TO M(I)
8310          T0=T0+SQR(((X0(J)-X0(J-1))/X2)^2+((Y0(J)-Y0(J-1))/Y2)^2)
8320        NEXT J
```

2.A.4 CALCULATE ADJUSTED DASHED LINE PARAMETERS

```
8330        M1=1 MAX INT(T0/SUM(P0)+0.5)
8340        P0=T0/(M1*SUM(P0))*P0
```

2.A.5 MOVE TO START OF CURVE

```
8350        J=1
8360        D1=X0(J)
8370        D2=Y0(J)
8380        MOVE @D:D1,D2
```

2.A.6 LOOP OVER M1 PATTERNS AND L ELEMENTS IN EACH PATTERN

```
8390        FOR I1=1 TO M1
8400          FOR J1=1 TO L
8410            D3=0
8420            D4=0
8430            REM
```

2.A.7 CALCULATE DISTANCE TO NEXT DATA POINT; IF GREATER THAN PATTERN ELEMENT, INTERPOLATE TO FIND WHERE TO GO AND SET FLAG TO INDICATE DONE WITH ELEMENT--IF LESS, GO TO NEXT DATA POINT

```
8440              D5=(X0(J+1)-D1)/X2
8450              D6=(Y0(J+1)-D2)/Y2
8460              D7=SQR(D5^2+D6^2)
8470              IF D4+D7=>P0(J1) THEN 8530
8480                J=J+1
8490                D1=X0(J)
8500                D2=Y0(J)
8510                D4=D4+D7
8520                GO TO 8590
8530              REM
8540                IF D7<=1.0E-4 THEN 8570
```

```
8550                 D1=D1+X2*(P0(J1)-D4)*D5/D7
8560                 D2=D2+Y2*(P0(J1)-D4)*D6/D7
8570               REM
8580               D3=1
8590             REM
```

2.A.8 IF EVEN-NUMBERED ELEMENT BEING PROCESSED, MOVE; IF ODD-NUMBERED, DRAW

```
8600             IF J1=2*INT(J1/2) THEN 8630
8610               DRAW @D:D1,D2
8620               GO TO 8650
8630             REM
8640               MOVE @D:D1,D2
8650             REM
```

2.A.9 TEST IF DONE WITH CURRENT ELEMENT

```
8660           IF D3<>1 THEN 8430
8670         NEXT J1
8680       NEXT I1
8690     REM
8700     GO TO 9180
```

2.B SOLID LINE (CURVE TYPE 4)

```
8710   REM
8720     D1=FNA(X(I,1))
8730     D2=FNB(Y(I,1))
8740     MOVE @D:D1,D2
8750     FOR J=1 TO M(I)
8760       D1=FNA(X(I,J))
8770       D2=FNB(Y(I,J))
8780       DRAW @D:D1,D2
8790     NEXT J
8800     GO TO 9180
```

2.C SYMBOLS (CURVE TYPES 5 THROUGH 9)

```
8810   REM
8820     FOR J=1 TO M(I)
8830       D1=FNA(X(I,J))
8840       D2=FNB(Y(I,J))
8850       GO TO L0(I)-4 OF 8860,8920,8980,9050,9110
```

2.C.1 CROSSES

```
8860       REM
8870         MOVE @D:D1+1.5*X3,D2
8880         DRAW @D:D1-1.5*X3,D2
8890         MOVE @D:D1,D2-1.5*Y3
8900         DRAW @D:D1,D2+1.5*Y3
8910         GO TO 9160
```

2.C.2 X-SHAPED

```
8920       REM
8930         MOVE @D:D1+X3,D2+Y3
8940         DRAW @D:D1-X3,D2-Y3
8950         MOVE @D:D1-X3,D2+Y3
8960         DRAW @D:D1+X3,D2-Y3
8970         GO TO 9160
```

2.C.3 SQUARES

```
8980       REM
8990         MOVE @D:D1-X3,D2-Y3
9000         DRAW @D:D1-X3,D2+Y3
9010         DRAW @D:D1+X3,D2+Y3
9020         DRAW @D:D1+X3,D2-Y3
9030         DRAW @D:D1-X3,D2-Y3
9040         GO TO 9160
```

2.C.4 TRIANGLES

```
9050       REM
9060         MOVE @D:D1,D2+2*Y3/SQR(3)
9070         DRAW @D:D1+X3,D2-Y3/SQR(3)
9080         DRAW @D:D1-X3,D2-Y3/SQR(3)
9090         DRAW @D:D1,D2+2*Y3/SQR(3)
9100         GO TO 9160
```

2.C.5 UPSIDE-DOWN TRIANGLES

```
9110       REM
9120         MOVE @D:D1,D2-2*Y3/SQR(3)
9130         DRAW @D:D1-X3,D2+Y3/SQR(3)
9140         DRAW @D:D1+X3,D2+Y3/SQR(3)
9150         DRAW @D:D1,D2-2*Y3/SQR(3)
9160       REM
9170     NEXT J
9180   REM
```

3. PRINT LEGEND FOR I'TH CURVE

```
9190   W$=SEG(U$,1,POS(U$,"#",1)-1)
9200   U$=SEG(U$,POS(U$,"#",1)+1,LEN(U$)-POS(U$,"#",1))
9210   D1=FNA(X(I,M(I)))
9220   D2=FNB(Y(I,M(I)))
9230   MOVE @D:D1+3*X3,D2-0.5*S3*Y2*(1.125*C4)
9240   PRINT @D:W$
```

4. PLOT CURVE FIT IF REQUIRED

```
9250   IF C=0 THEN 9280
9260     MOVE @D:U1,A(I)*U1+B(I)
9270     DRAW @D:U2,A(I)*U2+B(I)
```

```
9280   REM
9290 NEXT I

              RETURN PEN TO HOLDER AND EXIT

9300 PRINT @D,8:0
9310 HOME @D:
9320 RETURN

              INITIALIZATION ROUTINE

                  ARRAY K IS USED TO KEEP TRACK OF WHICH USER KEYS
                  HAVE BEEN DEPRESSED; LINE 9350 DEFINES THE ADDRESS OF
                  THE 4924 TAPE DRIVE

9330 DIM K(5),R$(1)
9340 R$=REP("",1,0)
9350 D0=2
9360 IF LEN(R$)>0 THEN 9390
9370 K=0
9380 R$=" "
9390 REM
9400 RETURN
```

DISTRIBUTION LIST
TR-368

No. of Copies	Organization
3	Commander US Army Materiel Development and Readiness Command ATTN: DRCDM-S DRCDE-SD DRCDE-SA 5001 Eisenhower Avenue Alexandria, VA 22333
2	Commander Defense Technical Information Center ATTN: TCA Cameron Station Alexandria, VA 22314
2	Chief Defense Logistics Studies Information Exchange US Army Logistics Management Center ATTN: DRXMC-D FT Lee, VA 23801
1	HQDA (SAUS-OR) WASH DC 20310
1	HQDA (DAMO-RQA) WASH DC 20310
2	Commander US Army Concepts Analysis Agency ATTN: CSCA-MR CSCA-ADL 8120 Woodmont Avenue Bethesda, MD 20014
1	Commander US Army Operational Test and Evaluation Agency ATTN: CSTE-STS-L 5600 Columbia Pike Falls Church, VA 22041
5	Commander US Army Armament Research and Development Command ATTN: DRDAR-SE DRDAR-LCS DRDAR-SCF DRDAR-SCS DRCPM-ADG Dover, NJ 07801
2	Commander US Army Aviation Research and Development Command ATTN: DRDAV-B DRDAV-E PO Box 209 St. Louis, MO 63166
1	Commander US Army Electronics Research and Development Command ATTN: DRDEL-ST-SA (A. Sambuco) 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Troop Support and Aviation Materiel Readiness Command ATTN: DRSTS-BA 4300 Goodfellow Boulevard St. Louis, MO 63120
2	Commander US Army Combat Developments Experimentation Command ATTN: ATEC-PL-M ATEC-PL-TL FT Ord, CA 93941
2	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-PL ATAA-TG White Sands Missile Range, NM 88002

DISTRIBUTION LIST (Continued)
TR-368

No. of Copies	Organization
1	Commandant US Army Air Defense School ATTN: ATSA-CD-A FT Bliss, TX 79916
1	Commander US Army Aviation Center and Fort Rucker ATTN: ATZQ-CD FT Rucker, AL 36362
1	President US Army Air Defense Board ATTN: ATZC FT Bliss, TX 79916
2	Commander US Army White Sands Missile Range ATTN: STEWS-TE-ML STEWS-NR-PA White Sands Missile Range, NM 88002
1	Commander US Army Tank-Automotive Command ATTN: DRDTA-V Warren, MI 48090
1	Commander US Army Missile Command ATTN: DRSMI-DS Redstone Arsenal, AL 35809
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-IEP-L Rock Island, IL 62199
1	AFELM, The Rand Corporation ATTN: Library-D 1700 Main Street Santa Monica, CA 90406
1	Tektronix, Inc. Information Display Division Applications Library Group 451 P.O. Box 500 Beaverton, Oregon 97077
	Aberdeen Proving Ground
3	Cdr, USATECOM ATTN: DRSTE-CT DRSTE-AD DRSTE-CM
1	Dir, HEL ATTN: DRXHE-SP
2	Dir, BRL ATTN: DRDAR-BLB DRDAR-TSB-S (STINFO)

DISTRIBUTION LIST (Continued)
TR-368

Number of Copies	Organization
46	Dir, AMSAA ATTN: DRXSY-A (D. O'Neill) DRXSY-AD (H. Peaker) DRXSY-AA (T. Coyle) DRXSY-ADG (J. Meredith) DRXSY-ADG (D. Drake) DRXSY-ADG (J. Foulkes) DRXSY-ADG (K. Frounfelker) DRXSY-ADG (B. Lufkin 5 cys) DRXSY-ADG (R. Sebra) DRXSY-ADG (J. Wald) DRXSY-AAM (J. Passamonte) DRXSY-AAM (P. Hill) DRXSY-AAM (J. Hennessey) DRXSY-AA (T. Coyle) DRXSY-GI DRXSY-GS DRXSY-GA DRXSY-GB DRXSY-GP DRXSY-CC DRXSY-CS DRXSY-CM DRXSY-CR DRXSY-CI DRXSY-CA DRXSY-RW DRXSY-RE DRXSY-RV DRXSY-RM DRXSY-FA DRXSY-FX DRXSY-FM DRXSY-FR DRXSY-FO DRXSY-FLSO DRXSY-FIRO DRXSY-P DRXSY-J DRXSY-DA DRXSY-PRO DRXSY-HEL DRXSY-A (Editorial Ofc)